Keyword Location in Noisy Document Images

Siamak Khoubyari and Jonathan J. Hull
Department of Computer Science
Center of Excellence for Document Analysis and Recognition
State University of New York at Buffalo
Buffalo, New York
USA
hull@cs.buffalo.edu
khoub-s@cs.buffalo.edu

Abstract

It may be difficult to locate keywords in noisy document images because of degraded OCR performance. A new technique for word image matching has the potential to select those word images in a document that represent potential keywords and to generate improved prototypes for those keywords. No explicit recognition is performed in this process, but better OCR performance will occur on the improved prototypes than would occur on any of the isolated words. The proposed method for keyword selection and recognition is best suited for document indexing in an image-based document retrieval system. This paper presents an algorithm for word image clustering and discusses how it is applied to locate groups of equivalent word images in a document. Improved prototypes are generated for clusters that represent potential keywords. The results of applying the algorithm to an article in the Brown Corpus are given. The keywords chosen by this approach and those chosen from the ASCII text of the article by a conventional keyword selection methodology are compared. The potential for improvement in recognition performance on those keywords is also demonstrated.

1 Introduction

Text recognition algorithms convert images of text into their ASCII equivalent. This process is becoming an increasingly important research area as the number of documents that need to be converted increases and the difficulties that currently available OCR (optical character recognition) devices can have in recognizing the text in those documents becomes clear [1]. The lack of robust OCR capability is especially apparent for noisy document images that contain touching or broken characters.

Text recognition algorithms have typically relied on the segmentation of words into isolated characters followed by recognition of the character images. Sometimes the character decisions are postprocessed versus a dictionary of allowable words. Such techniques effectively use the redundancy between the letters of legal words to improve performance. An extension of this strategy to whole word recognition uses the context of character occurrence within word images directly in the recognition process [2].

The performance of both techniques deteriorates in the presence of noise that de-

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

grades the outline shape of characters or causes characters to touch one another. These types of noise occur in facsimile images and photocopier output, two of the more common application domains. Word recognition methods have been successful in improving performance to some extent [3]. However, they still require large dictionaries to be comprehensive.

Knowledge about the language in which a document is written can be used to improve text recognition performance. Language-level knowledge sources that have been utilized have included semantics of chess games [4] as well as the statistical constraints between part-of-speech tags [5]. Another useful characteristic of language is the frequency of word occurrence. This has been used for solving substitution ciphers and OCR processes [6]. Two especially useful word occurrence characteristics are the frequency of *function* words and the internal frequency of *content* words within a document.

While function words are short determiners or prepositions that supply syntactic information about nearby words, content words are usually nouns that convey information about the topic of an article. Often, the same content words recur several times within an article. This effect is utilized in document classification and information retrieval techniques that select keywords based on their frequency. Often the first step in locating a keyword (index term) is to discard the most frequent words, which tend to be function words, and choose the words from the remaining set that have high internal frequency within the document [7]. These methods have also been extended to phrasal indexing where repeated groups of words are used to improve the effectiveness of document classification.

This paper proposes a technique that uses the repetition of words within a document to improve the recognition of important content words. Groups of *equivalent* word images are created by matching images to one another – no explicit recognition is applied. A result of this approach is that touching, broken, or degraded characters that would render typical recognition algorithms useless are easily compensated for by inter-word redundancy. The image clusters containing potential content words are then located. Improved prototypes are generated for the content words using inter-image redundancy to eliminate noise. These prototypes could be used to improve the performance of a subsequent recognition system.

A further advantage of the content word location methodology is its potential usefulness in an image-based document classification and retrieval system. The improved images of content words could be passed to a recognition process. This would allow for automatic document classification techniques based on keyword and phrase recognition that would tolerate high degrees of noise in the source image. Many other words in a passage of text could be completely ignored and only the content words could be recognized. The processing efficiency of such an approach would be significant and the recognition accuracy would be high. A document retrieval system would then return images of the original documents to the user.

The rest of this paper presents a detailed statement of the word matching algorithm, including the word equivalence metric and an iterative clustering algorithm that uses the global features of word images to restrict processing time. A method for using the contents of the word image clusters to generate improved prototypes for recognition is also discussed. The ability of the proposed methodology to operate in the presence of noise is demonstrated by experimentation on degraded documents. Furthermore, the accuracy of the content word location results is determined by comparing them to the results from a conven-

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

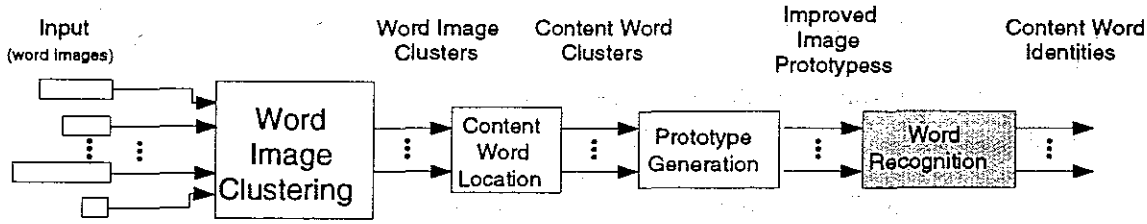

Figure 1: The major components of the algorithm

tional *automatic indexing* algorithm. Finally, the improvement in word recognition results using the prototype generation process is demonstrated on two high-end commercial OCR devices.

2 Keyword Location in Document Images

As illustrated in figure 1, the content word location and recognition algorithm has four main components. First, word images from an entire sample of printed text are segmented, and sent to a *clustering* algorithm to determine groups of equivalent word images. The ideal result is a number of clusters, each containing all the occurrences of a specific word in the document. It is important to note that the clustering step does *not* perform any recognition, as it only locates groups of *equivalent* words.

Using the characteristics of the clusters, and the word images within them, a *content word location* step then determines which of the clusters could contain content words. Using the redundancy between images in the clusters, clean prototypes for the content words can be generated, even if the individual words are moderately degraded. A word recognition algorithm can then yield better results when applied to these improved word prototypes.

The following sections explain these components in more detail.

2.1 Word Image Matching and Clustering

2.1.1 Clustering

The clustering algorithm uses an iterative process to locate and group equivalent word images. For each word image in the sample, the algorithm attempts to locate the cluster that contains equivalent words and adds the image to that cluster. If such a cluster does not exist, a new cluster is created. By the end of the process, the distinct words in the input document should have been placed in separate clusters. The main steps of the clustering algorithm are outlined in figure 2.

To locate the most similar cluster for a given word image, it would be intuitive to compare an input word image against *every* cluster in the entire cluster space. This, however, would result in an algorithm with $O(n^2)$ complexity. The algorithm used here heuristically reduces the computational cost by comparing each word to a limited number of clusters. This is done by using the global features of a word to partition the cluster space into regions, and comparing an input word only to clusters in similar regions. Examples of global features are the height and the width of the word image. This process is based on the reasonable assumption that two equivalent words that occur in the same document and are printed in the same font and size will most likely have similar (if not exactly equal) global

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

```
for each word image in a text passage do
    extract its global features
    locate the corresponding cluster-space region
    while more clusters exist in the region do
    {compare the word with each cluster in the region}
        if WordsEquiv (word image, cluster center) then
          add input word to the cluster
          exit while loop
        end if
    end while
    if equivalent cluster was not found then
        create a new cluster for the word
        link it to nearby equivalent clusters
    end if
end for
```

Figure 2: The top level clustering algorithm

features.

Once the region in the cluster space that contains the word image is identified, the input word is compared to the clusters in that region. If an equivalent cluster is located, the word is added to it, and the cluster is updated accordingly. Otherwise, a new cluster for that word is created in the region. Also, to account for equivalent clusters with slightly different global features, a search for equivalent clusters in the neighboring regions is conducted. This process continues until all the word images in the document have been processed.

2.1.2 Word Image Equivalence

An autocorrelation measure is used to determine whether two word images are equivalent. This is similar to the technique used in [8]. The value of the distance measure determines whether the words are equivalent. The three main steps in the process, as illustrated in figure 4, are outlined in figure 3.

The upper and lower baselines refer to the lines just above and below the lowercase letters that have neither ascenders nor descenders. The estimation of the baselines in a word is performed by finding a horizontal band of fixed height that covers the most dense area of the image. This area will approximately represent the "middle" region of the word.

Once the estimation is performed, the word images are aligned and centered on the lower baselines. The *exclusive-or* (XOR) between two images is then calculated and returned as a separate image. The resulting image contains black regions only where the two images differ. Just by counting the number of resulting black pixels in the XOR image, a measurement of similarity between the two images could be obtained. However, as explained below, this may not be sufficient for degraded images.

Figure 5 shows the result of calculating the exclusive-or between two pairs of images. The first involves two words with different identities, hence the process re-

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

```
WordsEquiv (Word_1, Word_2)
    CurrentDist ← ∞
    RelativePositions ← { UP, DOWN, RIGHT, LEFT, NONE }
    estimate baselines for Word_1, Word_2
    align Word_1, Word_2
    for each shift in RelativePositions do
        TempImage ← xor (Word_1, Word_2, shift)
        TempDist ← normalize (DistMap (TempImage))
        CurrentDist ← min (TempDist , CurrentDist)
    end for
    if CurrentDist < Threshold (Word_1, Word_2) then
      return (TRUE)
    else
      return (FALSE)
    end if
```

Figure 3: The word image matching algorithm

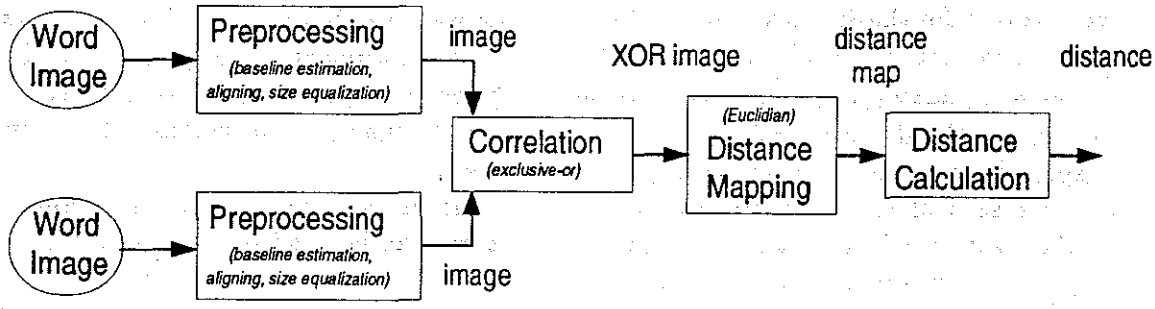

Figure 4: The main steps in word autocorrelation

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

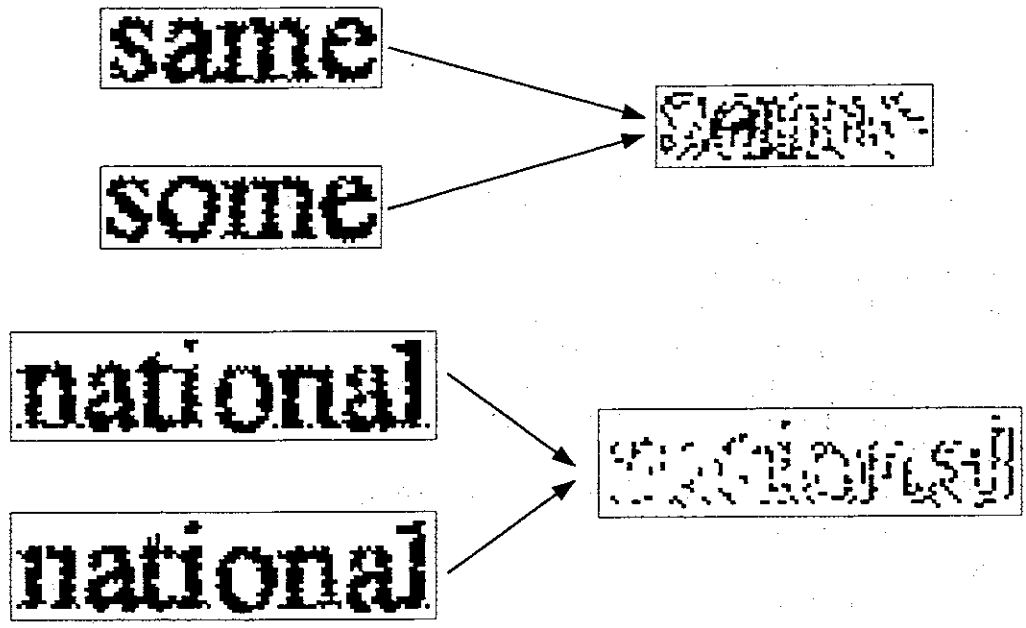

Figure 5: Exclusive-or results for sample word pairs

sults in an image that has large black areas ("blobs"). The second example shows two words that are equivalent, but contain a moderate amount of noise. In each case, if the distance were determined by simply counting the number of black pixels in the XOR image, the result would be very similar for both pairs of images. This illustrates the necessity for the distance measure to appropriately account for blobs.

Such an analysis can be performed by calculating the *distance map* for the XOR image [9]. A distance mapping algorithm replaces each black pixel with its distance to the nearest white pixel. The interior pixels in blobs are assigned higher values than those close to a boundary. Using the distance map, random noise can be distinguished from dense black regions by calculating the sum-of-the-squares of all the values in the distance map. This value is then normalized by the dimensions of the two images. The result is higher distances for XOR images that contain dense blobs, such as the ones in the top part of figure 5, and lower values for the less significant black regions such as the ones shown in the second part of the figure.

2.2 Content Word / Phrase Location

Content words are those that add meaning to the document. It is possible to form a general notion of the content of a given document, simply by inspection of its content words.

In an information retrieval system, *keywords* can be used to restrict the set of retrieved documents, according to their contents. That is, each on-line document is classified by the keywords it has been assigned. Although the best judgment for important keywords can be made by a human, the process of manual keyword selection is a time-consuming one, and can be very costly. Therefore, many automatic keyword selection algorithms have

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

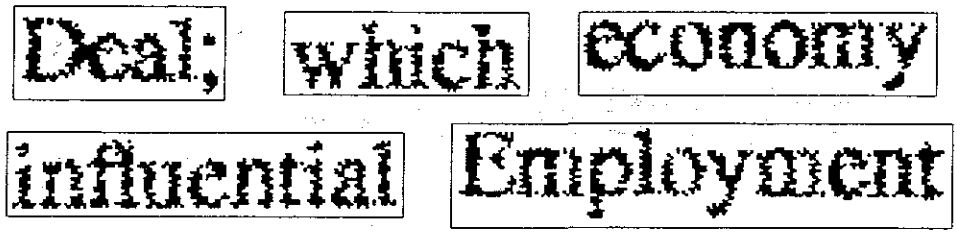

Figure 6: Instances of word images with moderate noise

been proposed [10]. These techniques use the machine-readable text of the document to determine the words that are descriptive index terms. In document analysis, this can be achieved by recognition of the document image, and then analysis of the resulting ASCII text. However, the limitations of the current OCR technology make this difficult when the text is even moderately noisy. The method described here does not rely on the results of recognition. Rather, keywords are located (without recognition) directly from the document image.

Content words are usually not as frequent as function words. Nevertheless, they tend to appear frequently within a given document [11]. This fact, along with the observation that content words tend to be longer than other words, can help in locating many of these words. That is, the clusters of content words contain many instances of relatively long words. The significance of the number of words in a cluster can be determined by comparison to a threshold determined by the number of words in the document. In this way, many of the important keywords within a document can be located without performing any recognition.

Location of potential content words can be extended to the location of important *phrases* in a document. Just as in the case of semi-frequent (longer) words being important, two- or three-word phrases that occur multiple times in the sample can potentially provide valuable information about the subject of the document. To locate such phrases, transitions among clusters are observed. That is, since the words are clustered in the sequence that they appear in the text, the clusters into which the k previous words were placed are known. This way, the transitions between any two or three clusters can be noted *while* the clusters are being formed. At the end of this process, the most frequent two- and three-cluster transitions are picked, thus identifying the most frequent two- and three-word phrases in the document.

2.3 Prototype Generation

The presence of noise such as touching or broken characters has always hampered the operation of OCRs. Given only one word which contains even moderate amounts of such noise (fig. 6), it is sometimes difficult to recognize the word using isolated character recognition techniques.

One of the advantages of clustering equivalent word images is that a recognition algorithm is provided with several copies of the same word. Using those multiple examples, a prototype can then be generated for that word that would yield better recognition performance than would be possible with any of the original (degraded) word images. One possible method of building such prototypes is explained below.

The equivalent word images are first aligned on their estimated baselines, and used to construct the corresponding pro-

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

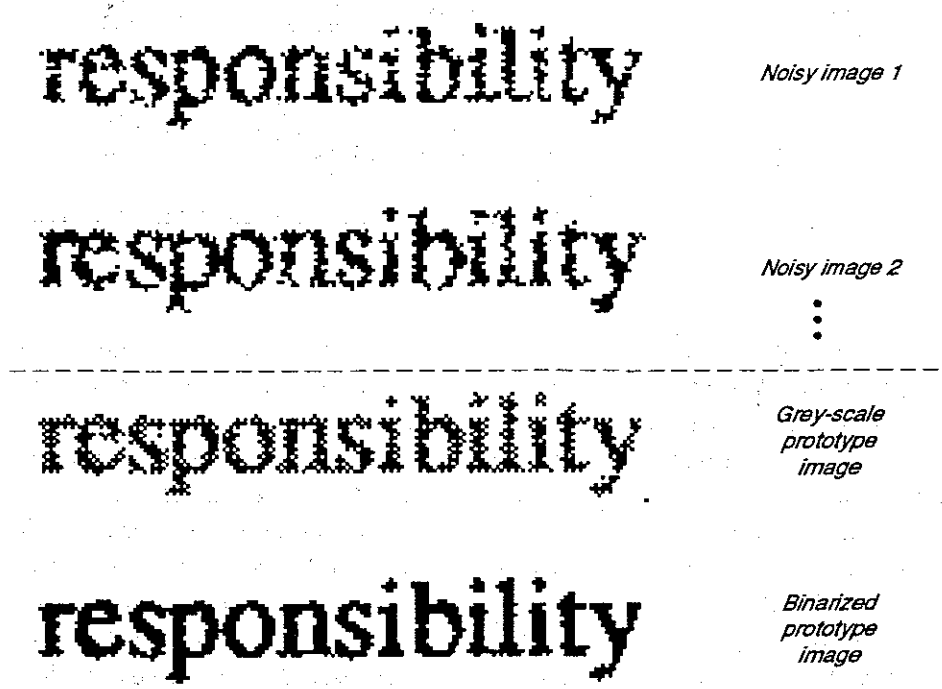

Figure 7: An example of the prototype generation process

totype image as follows. Each pixel position in the prototype image is assigned the likelihood of that pixel being black in the clustered set of images. The result is a grey-scale prototype image, where the darkest regions represent the areas of the word with the highest probability of being black. By binarizing the image using a normalized threshold, a relatively clean prototype image can be derived (fig. 7).

The advantages of this technique are obvious. The availability of such improved-quality word images could facilitate any recognition process (compared to the original noisy images). As shown by the example in figure 7, the instances of broken characters, and the general noise around the edges of a word are reduced. The performance of most word recognition techniques would be improved by this process since the probability of successful feature extraction would be increased.

Another advantage of the prototype generation process is in character segmentation. It is possible that the different instances of a given word in a document contain touching characters in *different* places. Therefore, by more detailed analysis of the grey-scale prototype image, it may be possible to derive reliable character-segmentation points, facilitating the use of character-based recognition methods.

3 Term Weighting

As mentioned in the previous section, many automatic indexing algorithms have been developed which determine some measure of importance for each word in the document. One such *single-term* automatic indexing algorithm [11] uses the internal frequency of words within a document, as well as the number of documents a word occurs in, to determine the weight for each poten-

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

code	category	samples	code	category	samples
A	Press: Reportage	44	J	Learned & Scientific	80
B	Press: Editorial	27	K	Fiction: General	29
C	Press: Reviews	17	L	Fiction: Mystery & Detective	24
D	Religion	17	M	Fiction: Science	6
E	Skills & Hobbies	36	N	Fiction: Adventure & Western	29
F	Popular Lore	48	P	Fiction: Romance & Love Story	29
G	Belles Letters	75	R	Humor	9
H	Miscellaneous	30			

Table 1: The 15 *genres* in Brown Corpus and the number of samples in each

tial index term. A version of this algorithm has been implemented for the purposes of comparison to the method proposed in the previous section. Although many other indexing algorithms have been proposed, the chosen technique has been shown to outperform more elaborate systems in most cases [11].

Once the text is read, those words in the document that also appear in a *stop list* [12] of frequent function words are removed, as they are not content-bearing. The remaining words in the document are then analyzed and *weighted* as follows.

Two characteristics of the words are considered in calculating the weight for each term: the *internal term frequency* (tf) and the inverse of the *document frequency* (df). The former simply refers to the frequency of the word within the given document. The latter measure refers to the number of documents in the collection that contain the word in question. This tests whether the word is concentrated in only a few documents, or whether it is distributed randomly throughout the collection. The weight w_j of the j^{th} term in the document is then computed using the formula:

$$w_j = tf_j \times log\,(N/df_j)$$

where N is the number of documents in the collection. The words in the document can then be sorted according to their weight, and the top index terms chosen.

4 Experimental Results

4.1 Experimental Environment

4.1.1 Text Image Generation

It is well-known that different styles of writing are used, depending on the subject matter and the author. Since the method proposed in this paper uses linguistic information, experimentation should include sufficiently large passages of *running* text from many different sources and styles. Furthermore, the selection of the text used in any study should be close to "real-life" in both appearance and content.

The Brown Corpus is an *on-line* collection of over one million words of actual excerpts from printed material published in the year 1961. It was carefully designed by Kučera and Francis [13] at Brown University to reflect modern "edited American English." This resulted in the division of the corpus, proportionally, into 15 subject categories, ranging from Scientific to Press Reportage to Humor. The resulting composition of the corpus is shown in table 1.

The name of each category in table 1 is followed by the number of "samples" it includes, where each sample contains approximately 2000 words of running text.

The on-line text of any article in the corpus can be used to derive the index term weights for the keywords it contains. Using a postscript-to-bitmap conversion process,

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

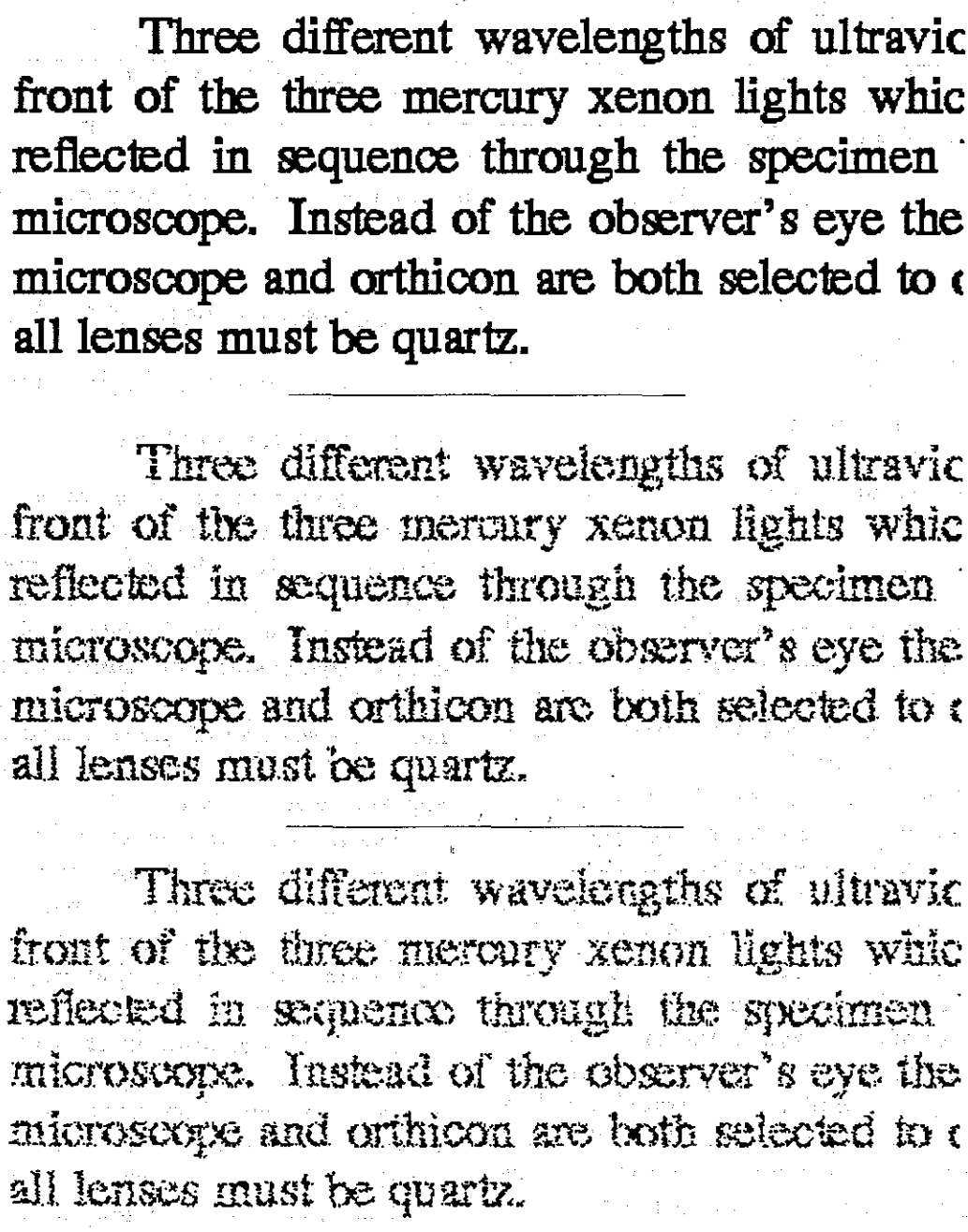

Figure 8: A "clean" image segment and two degraded versions generated by two iterations with σ 30,30(middle) and 50,60(bottom)

the ASCII text can also be used to generate the corresponding word images, that are then used to demonstrate the algorithm presented here.

4.1.2 Noise Generation

To experiment with degraded text, an artificial noise generation process is applied to the word (or page) images to simulate some of the noise present in photocopies and facsimiles.

First, more instances of touching characters are created by thickening the stroke width of the characters. Then a modified Gaussian noise process is used to set random black pixels on the letter *boundaries* to white. This generates ragged letter shapes and broken characters depending on the noise level. The amount of such noise can be controlled by the standard deviation (σ), and the number of times the process is repeated. A high value for the standard deviation means more boundary pixels will be removed. By iterating the process of boundary degradation, even more degraded images can be generated. Since obtaining many copies of noisy document images may be difficult, this method provides an acceptable, consistent alternative for producing degraded images, thus allowing for extensive experiments. An examples of the application of the noise generation process can be seen in figure 8.

4.2 Content Word Location

The experiments reported here were performed on several samples of the Brown Corpus, each containing about 2000 words. The sample numbers and the corresponding article titles are shown in table 2.

Clustering and content word location experiments were performed on word images produced from these samples. Separate experiments were completed using two different noise levels, illustrated by the text sample shown in figure 8.

Good clustering performance was achieved in all cases. This in turn translated into more reliable results from the content word location algorithm, as displayed in table 3. The table shows, for each sample, the identity of some of the image clusters which were located as containing potential content words. Given the theme of each article, the results illustrate that the content word location algorithm captures the image clusters for many of the words that have direct bearing on the subject matter (e.g. for sample G02: *responsibility, concept, national, sovereignty*).

In order to validate the results from the content word location algorithm, they were compared to the results from a conventional automatic indexing algorithm. The term weighting algorithm described in section 3, which is similar to an algorithm proposed in [11], was implemented and tested on the same Brown Corpus samples. This process has access to the ASCII text of the samples in question, as well as the entire online corpus. The top 10 index terms chosen from each article, and their weights, are shown in table 4.

The performance of the content word location algorithm described in this paper can be measured by calculating the overlap between the two lists of keywords. For the four test samples used here, 76.2% of the top 20, and 92.5% of the top 10 *index terms* which were selected from the ASCII text were correctly located as *content words* in the corresponding document image.

4.3 Content Word Prototype Recognition

To demonstrate the effectiveness of the prototype generation technique, an experiment using 100 of the content word clusters was performed. Three word images, representing different instances of the same word, were taken from each of the 100 content

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

sample	title, author, source, year
E25	*Advances in Medical Electronics*, by W.H. Buchsbaum, Electronics World, 1961
G02	*Toward a Concept of National Responsibility*, by A.S. Miller, The Yale Review, 1961
J61	*Completing and Restoring the Capitol Frescos*, by A. Cox, Museum News, 1961
J74	*Food Preservation by Ionizing Radiation*, by H.W. Nelson, Battelle Technical Review, 1961

Table 2: The Brown Corpus samples used in experiments.

E25	G02	J61	J74
electronics	responsibility	Brumidi	foods
ultraviolet	nationalism	frieze	irradiation
medical	world	fresco	preservation
transducer	sovereignty	Costaggini	meats
patient	principle	Capitol	spoilage
ultrasonic	concept	painting	radiopasteurization
technique	American	plaster	refrigeration
blood	government	sketches	palatability
radiation	Congress	cleaning	microorganisms
microscope	political	Rotunda	ionizing

Table 3: Some of the content words located from each sample.

E25		G02		J61		J74	
Word	Weight	Word	Weight	Word	Weight	Word	Weight
electronics	51.8	sovereignty	43.2	Brumidi	47.5	food	79.7
ultraviolet	47.5	responsibility	42.7	frieze	43.2	foods	76.1
heart	38.6	nationalism	35.2	lime	43.2	radiation	64.5
ultrasonic	34.5	principle	24.2	Costaggini	30.2	irradiation	43.2
medical	32.5	nations	22.2	Brumidi's	25.9	meats	38.9
sonar	30.2	concept	20.2	fresco	25.9	palatability	38.9
transducer	30.2	national	17.1	Capitol	21.6	preservation	38.9
electronic	25.9	internal	15.2	cleaning	18.1	radiopasteurization	38.9
pill	25.9	territorial	14.7	plaster	17.3	refrigeration	38.9
blood	25.8	welfare	11.9	sketches	17.3	doses	34.5

Table 4: Top 10 keywords for each sample determined by automatic term weighting

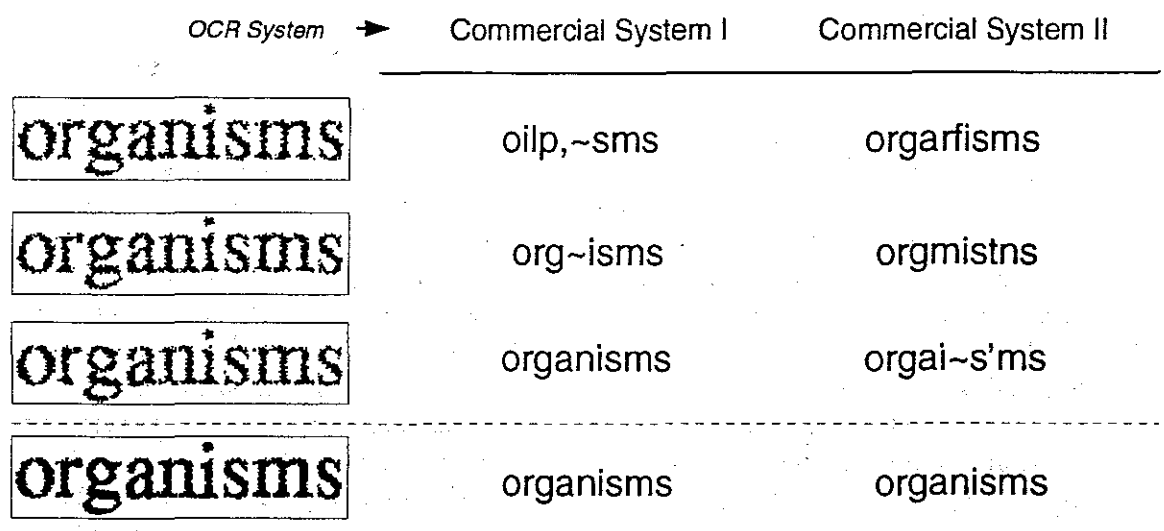

Figure 9: An instance of OCR results on original noisy images, and the generated prototype

OCR system	noisy word recognition	prototype word recognition
I	84%	96%
II	82%	99%

Table 5: Overall OCR results

word clusters. Also included were the corresponding improved prototype images, which were generated using the process described in this paper. Each word image was passed to two high-end commercial OCR devices and their word recognition performance was determined.

Figure 9 shows the OCR results for a single keyword[1]. The first three word images are degraded originals which were members of the corresponding content word cluster, while the fourth word image in figure 9 is the generated prototype for that cluster.

Table 5 shows the overall results for the test images. For each of the two OCR systems, the percentage of words recognized correctly are shown both for the noisy originals and the improved prototypes. A word is considered correct if all its characters were recognized correctly. The table shows an average of 14.5% improvement in word recognition rates by using the generated prototypes.

4.4 Discussion and Enhancements

There is a reasonable correlation between the content words located using the word image clustering results, and the index terms found by applying the term weighting function to the ASCII text. It is believed that the performance can be further improved by incorporating the following enhancements in the algorithm.

In addition to the internal frequency of the words, the term weighting function also uses the inverse document frequency – the number of *other* documents the word appears in. This idea can also be integrated into the content word location process. The list of content words could be filtered by

[1]Since the latest versions of the OCR devices were still in pre-release form, the manufacturers are not identified

Symp. on Document Analysis and Information Retrieval, Las Vegas, NV, April 26-28, 1993.

lowering the weight of those words that appear in many other documents in the collection. To do this, the word images can be matched to similar word image clusters that were formed for the other documents. If a match is found, the inverse document frequency for that content word would be decreased. The result would be a new ranking for the list of content words.

Another possible enhancement to the algorithm stems from the observation that many content words which appear in the document are *nouns*, and hence are part of a noun phrase, possibly containing one or more *function* words (e.g. *"of the people", "in the Congress",...*). Just as potential content word clusters can be located and recognized, those containing frequent function words can also be located [14]. Realizing that a potential content word is a noun by locating it within a noun phrase can also help to increase its weight.

Content *phrase* location can also help in understanding the document. A frequent phrase can be located in the document by observing the large number of transitions between the respective clusters. That is, if the words in one cluster are frequently adjacent to the words in another cluster, those words form a frequent phrase. Higher order transitions between two clusters that surround a shared "adjacent" cluster, can also be considered when locating frequent phrases. The combination of these two techniques could be used to locate important phrases such as "*national responsibility*" and "*food preservation*", without recognizing the words.

5 Conclusions

The promising results from the content word location process show that the words pertaining to the content of a document can be located even *before* their recognition. The prototype generation step could then improve the recognition performance on those words. This means that by locating and recognizing a few words, the subject of a document could be determined, possibly making it unnecessary to perform recognition on the rest of the document, which may be very degraded. It was demonstrated that by generating improved prototypes for noisy content word images, better OCR performance can be achieved.

This process is suited for application to an image-based document retrieval system in which documents are characterized by the keywords they contain. First, the content word clusters would be passed through the prototype generation process to produce improved images for the keywords. A segmentation or word recognition program would then recognize the improved images, thereby determining the identity of the keywords. Documents would then be classified by those keywords and the document images would be placed on secondary storage. Queries would be processed by comparison to the keywords and images of the relevant documents returned to the users.

Acknowledgment

We would like to thank Dr. Junichi Kanai of the Information Science Research Institute at the University of Nevada at Las Vegas for arranging the OCR tests.

References

[1] Roger Bradford and Thomas Nartker. Error correlation in contemporary OCR systems. In *First International Conference on Document Analysis and Recognition*, pages 516–524, Saint-Malo, France, 1991.

[2] Jonathan J. Hull. Hypothesis generation in a computational model for vi-

sual word recognition. *IEEE Expert*, 1(3):63–70, 1986.

[3] Tin Kam Ho, Jonathan J. Hull, and Sargur N. Srihari. Combination of structural classifiers. In *IAPR Workshop on Syntactic and Structural Pattern Recognition*, Murray Hill, New Jersey, June 13-15 1990.

[4] H. S. Baird and K. Thompson. Reading chess. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-12:552–559, 1990.

[5] Jonathan J. Hull. A hidden markov model for language syntax in text reconition. In *11th IAPR International Conference on Pattern Recognition*, The Hague, The Netherlands, August 30 - September 3 1992.

[6] George Nagy, Sharad Seth, and Kent Einsphar. Decoding substitution ciphers by means of word matching with application to OCR. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5):710–715, September 1987.

[7] Gerard Salton. Developments in automatic text retrieval. *Science*, 253:974–980, 1991.

[8] Norman F. Brickman and Walter S. Rosenbaum. Word autocorrelation redundancy match (WARM) technology. *IBM Journal of Research and Development*, 26(6):681–686, November 1982.

[9] Per-Erik Danielsson. Euclidian distance mapping. *Computer Graphics and Image Processing*, 14:227–248, 1980.

[10] C.J. van Rijsbergen. *Information Retrieval.* Butterworths, London, 1979.

[11] Gerard Salton. *Automatic Text Processing.* Addison-Wesley, 1988.

[12] Christopher Fox. A stop list for general text. *SIGIR Forum*, 24:19–35, 1989.

[13] Henry Kučera and W. Nelson Francis. *Computational Analysis of Present-Day American English.* Brown University Press, Providence, Rhode Island, 1967.

[14] Jonathan J. Hull, Siamak Khoubyari, and Tin Kam Ho. Word image matching as a technique for degraded text recognition. In *11th IAPR International Conference on Pattern Recognition*, volume 2, The Hague, The Netherlands, August 30 - September 3 1992.