Algorithms for Postprocessing OCR Results with Visual Inter-Word Constraints

Tao Hong and Jonathan J. Hull*
Center of Excellence for Document Analysis and Recognition (CEDAR)
State University of New York at Buffalo
Buffalo, New York 14228-2567
taohong@cs.buffalo.edu

RICOH California Research Center*
2882 Sand Hill Road, Suite 115
Menlo Park, CA 94025
hull@crc.ricoh.com

Abstract

Algorithms are presented that determine the visual relationships between word images in a document. These include instances of common word images and common substrings that occur often in English language text images. This information is then be used to improve the performance of a commercial optical character recognition (OCR) algorithm. The algorithms presented here calculate clusters of equivalent word images as well as common initial and final substrings. Experimental results are presented that show a 40% reduction in word level error rate is achieved on a test set of documents degraded by uniform noise.

1 Introduction

Typical commercial OCR systems recognize images of words by segmenting them into isolated characters and recognizing those images. A decision for a word image is represented as the concatenation of the decisions for its individual characters. A postprocessing algorithm is sometimes used that compares a string of character decisions to a word list and attempts to correct errors by finding the word that closely matches the string of character decisions.

A recent development has been to improve OCR postprocessing by considering the visual relations between words that exist in normal English language documents [1]. These relations include repeated words (such as "the", "of" and so on) and repeated substrings (for example, the word "the" occurs at the beginning of the word "theatre"). This recent work capitalized on this characteristic by requiring that the decisions output by a postprocessing algorithm for two different word images must be the same when those images contain common sequences of image data. That is, the symbolic decisions are the same when the corresponding image data is equivalent.

This approach takes advantage of the fact that it is often possible to determine that two word images or portions of word images are the same even when those images are degraded by noise that significantly impairs OCR performance. This same effect has been used previously in a method that determines the font and recognizes the function words in a document [3].

This paper presents algorithms that are used to calculate both whole-word as well as partial word relations between word images in a document image. This information is then used to constrain the decisions of an OCR so that they are the same when the corresponding portions of the images have been found to be equivalent.

2 Visual Inter-Word Relations

Word images from a page of text are related to each other by the six relations defined in Figure 1. Examples of those relations are shown in Figure 2.

Relation number one describes two images that are equivalent. This relation occurs often in normal English text where the same word is used many times in a single passage. Relation number two defines the occurrence of a *subimage*. That is, one word is entirely

0-8186-7310-9/95 $4.00 © 1995 IEEE

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00 © 1995 IEEE

	Possible Relations between W_1 and W_2	
	at image level	at symbolic level
1	$W_1 \approx W_2$	$W_1 = W_2$
2	$W_1 \approx subimage\_of(W_2)$	$W_2 = X \bullet W_1 \bullet Y$
3	$left\_part\_of(W_1) \approx left\_part\_of(W_2)$	$prefix\_of(W_1) = prefix\_of(W_2)$
4	$right\_part\_of(W_1) \approx right\_part\_of(W_2)$	$suffix\_of(W_1) = suffix\_of(W_2)$
5	$right\_part\_of(W_1) \approx left\_part\_of(W_2)$	$suffix\_of(W_1) = prefix\_of(W_2)$
6	$subimage\_of(W_1) \approx subimage\_of(W_2)$	$W_1 = X_1 \bullet Y \bullet Z_1$ and $W_1 = X_1 \bullet Y \bullet Z_1$

Note: "$\approx$" means approximately match at image level; "$\bullet$" means concatenation.

Figure 1: Word relations at the image and symbolic levels.

contained in another. Relations three through five define the *left_part_of* and *right_part_of* relations. These occur often because of the use of common prefixes and suffixes. Relation number six defines the occurrence of a subimage from one word as the subimage of another.

3 Algorithms for Detecting Visual Inter-Word Relations

The algorithm that detects the six visual inter-word relations is composed of six steps. First, the whole-word equivalence relation is detected by an image clustering algorithm. The image prototypes from each cluster (i.e., the average of the word images in the cluster) are then compared in five separate steps that determine the other five relations (including the *subimage*, as well as the various *left_part_of* and *right_part_of* relations).

3.1 Whole-Word Clustering Algorithm

The whole-word clustering algorithm is described in Figure 3. An agglomerative technique is used in which each image is compared to the list of current clusters. If the current word is not sufficiently similar to any of the available clusters, a new cluster is started. This process is continued until all the words have been processed. After clustering, any two word images in the same cluster are defined to be equivalent and thus hold relation number one with each other.

The visual similarity between two binary images is calculated as described below. This metric is used in the word image clustering algorithm described in Figure 3 in the *if* statement where it is determined whether an image *matches* with a prototype. Let A and B be two $m \times n$ binary images. Inside an image, "1" and "0" denote "*black*" and "*white*" pixel respectively. We measure visual similarity between A and B as

$$r(A,B) = \frac{\sum_i^m \sum_j^n (A_{ij} \wedge B_{ij})}{\sum_i^m \sum_j^n (A_{ij} \vee B_{ij})}$$

where "$\wedge$" and "$\vee$" are *and* and *or* operators respectively. The higher the measurement r is, the better two images match. When two images A and B are slightly different in size, the similarity between them is defined by the maximal matching obtained if A is shifted over B. By setting a proper threshold r_0, it is defined that two images are visually equivalent if $r(A,B) > r_0$. Further heuristics on image size (rows and columns) as well as other visual characteristics such as projection histogram similarity are also used to suppress incorrect matches.

3.2 Finding Sub-pattern Relations

The other five visual relations are based on different types of sub-patterns. These relations are all detected by comparing portions of the prototypes from the clusters generated by the whole-word clustering algorithm. The algorithm that detects that one image is a *subimage* of another is presented in Figure 4. This is done by comparing the cluster prototypes for shorter words to longer words using the **IsSubImage** metric. If two clusters have the *subimage* relation to each other, the individual words in the clusters are marked with this information.

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00 © 1995 IEEE

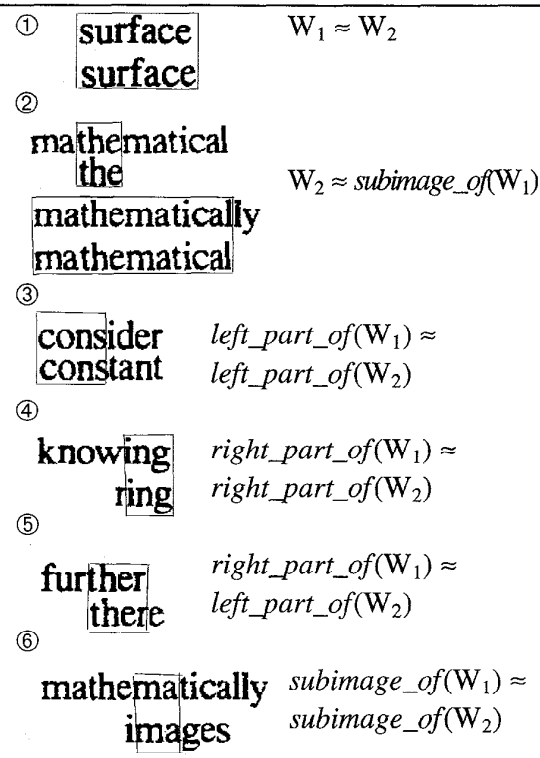

Figure 2: Examples of visual inter-word relations.

4 OCR Postprocessing With Visual Inter-Word Relations

A four-step algorithm is used to postprocess OCR results with visual inter-word relations [1]. The first three steps locate word decisions that are correct with high confidence. In the course of locating such high confidence decisions, some OCR errors are corrected. These high confidence word decisions are then used to learn images that correspond to individual characters and character sequences. These images are then used to decompose the remaining word images and generate new recognition results for them.

5 Experimental Results

An experimental system was developed to test the effectiveness of the algorithms presented above in improving OCR performance [1]. The input to this system is the output from a commercial OCR (i.e., Caere's AnyFont package) as well as the page images that were provided to the OCR. The commercial device provides at least a single decision for each word and in cases where it is unsure, several alternatives are produced. Also, the bounding box coordinates for each word are output.

Six page images containing over 5000 word images were used to test the system. These were scanned at 300 ppi and the binary image produced by the scanning hardware was used. Uniform noise was added to each image using the documentation degradation model (DDM) package from the University of Washington [2].

The accuracy of Caere's AnyFont OCR package on original pages is very high, more than 98% correct at the word level. After adding uniform noise with DDM, the word correct rate dropped to 73.5%. It was observed that the word alternatives produced by the OCR do not improve performance significantly.

Word clustering was then computed using the bounding boxes output by the OCR and inter-word relations were calculated between pairs of clusters. In the present implementation, only the first four visual

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00 © 1995 IEEE

```
extract all word images from the text image
set QUEUE as an empty set
put all word images into QUEUE
set IMAGE-CLUSTER-LIST as an empty set
while QUEUE not empty do
      extract an image I from QUEUE
      if image I matches with the prototype of a cluster in IMAGE-CLUSTER-LIST
        then
            add image I as a new member of that image cluster
        else if
            create a new image cluster with image I
            add it into IMAGE-CLUSTER-LIST
      end if
end while
```

Figure 3: Algorithm for word image clustering.

```
Word Image Clustering { see Figure 3 for detail }
sort clusters in IMAGE-CLUSTER-LIST by increasing order of the width of their prototypes
while IMAGE-CLUSTER-LIST has more than one cluster do
      extract a cluster C from IMAGE-CLUSTER-LIST
      for each cluster D in IMAGE-CLUSTER-LIST do
          if IsSubImage (prototype of C , prototype of D)
            then
                for each word image x in cluster C do
                    for each word image y in cluster D do
                        x is a subpattern of y
                    end for
                end for
          end if
      end for
end while
```

Figure 4: Algorithm for determining words that have the *subimage* relation.

relations in Figure 1 were used.

The overall word level correct rate was improved 86% to 92% and the accuracy of their alternative lists was improved from 87% to 95%. That is, the error rate was reduced by about 40%. It should be noted that the word images tested here account for about 71% of the words in the original text pages. Implementation of the rest of the algorithm will further improve performance.

6 Conclusions

In this paper several algorithms for computing visual relations between word images in a text document were presented. These relations include equivalence between whole words and partial equivalence between portions of words. A method for using these relations to improve the performance of a commercial OCR was discussed. Experimental results were presented that demonstrated the effectiveness of the postprocessing method.

References

[1] T. Hong and J.J. Hull, "Visual Inter-Word Relations and their Use in OCR Postprocessing," Third International Conference on Document Analysis and Recognition, Montreal, Canada, August 14-16, 1995.

[2] T. Kanungo, R. M. Haralick and I. Phillips, "Global and Local Document Degradation Models", *Proceedings of the Second International Conference on Document Analysis and Recognition ICDAR-93*, Tsukuba,Japan, 730-734, October 20-22, 1993.

[3] S. Khoubyari and J.J. Hull, "Font and Function Word Identification in Document Recognition," *Computer Vision, Graphics, and Image Processing: Image Understanding*, accepted to appear, 1995.

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00 © 1995 IEEE