A Hypothesis Testing Approach to Word Recognition Using an A^* Search Algorithm

Chi Fang and Jonathan J. Hull*
Center of Excellence for Document Analysis and Recognition (CEDAR)
State University of New York at Buffalo
Buffalo, New York 14228
chifang@cs.buffalo.edu

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

Abstract

An hypothesis testing approach for recognizing machine-printed words is presented in this paper. Based on knowledge of the document font and candidates for the identity of a word, this approach searches a tree of word decisions to generate and test hypotheses for character recognition and segmentation. The search starts at each sequential character position from both ends of a word image and proceeds inward. The accumulated cost of reaching a certain partial recognition decision is combined with the estimate of the potential cost to reach a goal state using an A^* search algorithm. The proposed algorithm compensates for local degradations by relying on global characteristics of a word image. Tests of the algorithm show a recognition rate of 98.93% on degraded scanned document images with touching characters.

Topic areas: hypothesis testing, A^* search algorithm, degraded document recognition, OCR

1 Introduction

Recognition of degraded words has been a difficult problem in OCR. A problem in recognizing degraded words is caused by the occurrence of touching characters and character fragmentation, which usually poses a difficulty for recognition algorithms that are based on character segmentation ([9, 2]). Algorithms have been proposed for recognizing degraded words that integrate segmentation with recognition and use recognition results to improve segmentation decisions ([10, 1, 12, 11]).

Another approach to word recognition is based on an analysis of word shape and on computation of *neighborhoods* of decisions for each word [6, 5]. The neighborhoods contain words that are visually similar to an input image. For example, the neighborhood for the word *word* might also contain the word *work* and the word *ward*. These algorithms effectively compensate for image noise by increasing the size of the neighborhood to guarantee that it contains the correct word. The reduction of a neighborhood to a single decision that matches the image is performed by a hypothesis testing algorithm [7].

In this paper a hypothesis testing approach is proposed that reduces the neighborhood for a given word image to a single choice. This method uses the neighborhood to generate a structured series of tests that are executed on the input image. The result is the word decision that best fits the input image. It is assumed that images of the individual characters in the font are available either through an explicit font learning step [8] or from a deciphering algorithm that first builds a font representation from a document [3].

For each word image, the hypothesis testing algorithm searches through a tree of word decisions and tests hypotheses for character segmentation and recognition at each sequential character position starting from both ends of the word image and proceeding inward. An A^* search algorithm ([4]) combines the accumulated cost of reaching a non-leaf node with the estimated cost to a leaf node in deciding which node to

0-8186-7128-9/95 $4.00 © 1995 IEEE

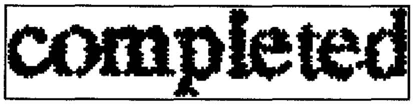
Proceedings of the Third International Conference on Document Analysis and Recognition (ICDAR '95)
0-8186-7128-9/95 $10.00 © 1995 IEEE

expand at each point in the search. Word and character shape characteristics that are relatively stable across image degradations are used for estimating the cost to reach a goal state (i.e., a word decision). This results in a search algorithm that exploits the heuristic power of visual constraints to speed up the search and to satisfy the admissibility of the A^* algorithm. The algorithm as a whole achieves precise word recognition by combining multiple local character segmentation and recognition choices and making the globally optimal decision using an A^* search. This provides a reliable and efficient recognition of degraded words.

The rest of this paper contains three sections. Section 2 contains a discussion of the major components of the proposed algorithm. Section 3 presents and discusses the experimental results of applying this hypothesis testing algorithm to a degraded document. Finally, conclusions and some future directions are pointed out in section 4.

2 Proposed Algorithm

Figure 1 shows a example of a degraded word image and its word candidate neighborhood provided by a preliminary word recognizer, ranked in ascending order of a value representing relative word mismatch. This plus the knowledge of fonts are the information available when the algorithm starts. We use this as an example to illustrate the ideas in the hypothesis testing algorithm.



completes	78756
composed	81052
strapped	81413
octagonal	82184
compared	82477
songbook	82964
completed	83449
comprised	83751
congested	83925
snapped	83956

Figure 1: Word image and its neighborhood.

For the word image and its candidate neighborhood in Figure 1, the algorithm starts by considering the left-most and right-most character positions in the word image. The word candidate set provides constraints as to what characters they could possibly be. For example, the left-most character could only be one of the characters from the set c, s, o, and the right-most character could only be one of the characters from the set s, d, l, k. The image prototypes for each of the characters in the two sets are then matched against the left-most portion and right-most portion of the word image. Each prototype match returns a value that measures the degree of character pattern match, as well the location in the word image where the best match was found.

The algorithm then evaluates the goodness of the matches at both ends of the word by averaging the two character image match values. The various combinations of characters that could occur at each end are considered. For this specific example, it is likely that the combination [c, d] will be ranked first, combination [c, l] ranked second, and so on.

In the next step, each potential combination of decisions at both ends is associated with a reduced candidate set. Combinations that have no candidate matching its character decisions at both ends of the current character position are discarded. In our example, after this stage the combination [c, d] will be associated with the reduced candidate set of *composed, compared, completed, comprised, congested*, the combination [o, l] will be associated with the reduced candidate set *octogonal*, and the combination [c, l] will be discarded because it has no candidate that matches the characters at both ends.

For each combination that remains, the algorithm generates a new state and associates it with the reduced candidate set and a new partial word image by cutting off the left-most and right-most portion of the current image that best match the character prototypes for this combination. All the new states generated by this process will be kept in a list together with a value that measures the overall goodness of the decisions made thus far.

The algorithm repeats the above process by choosing a state with the minimum cost from the list for further expansion. This process is continued until a word decision with minimum cost is found. That is, the cost for this decision should be less than the cost of reaching a goal state for all the alternatives on the list An example application of the proposed hypothesis testing algorithm is illustrated in Figure 2 by a tree structure.

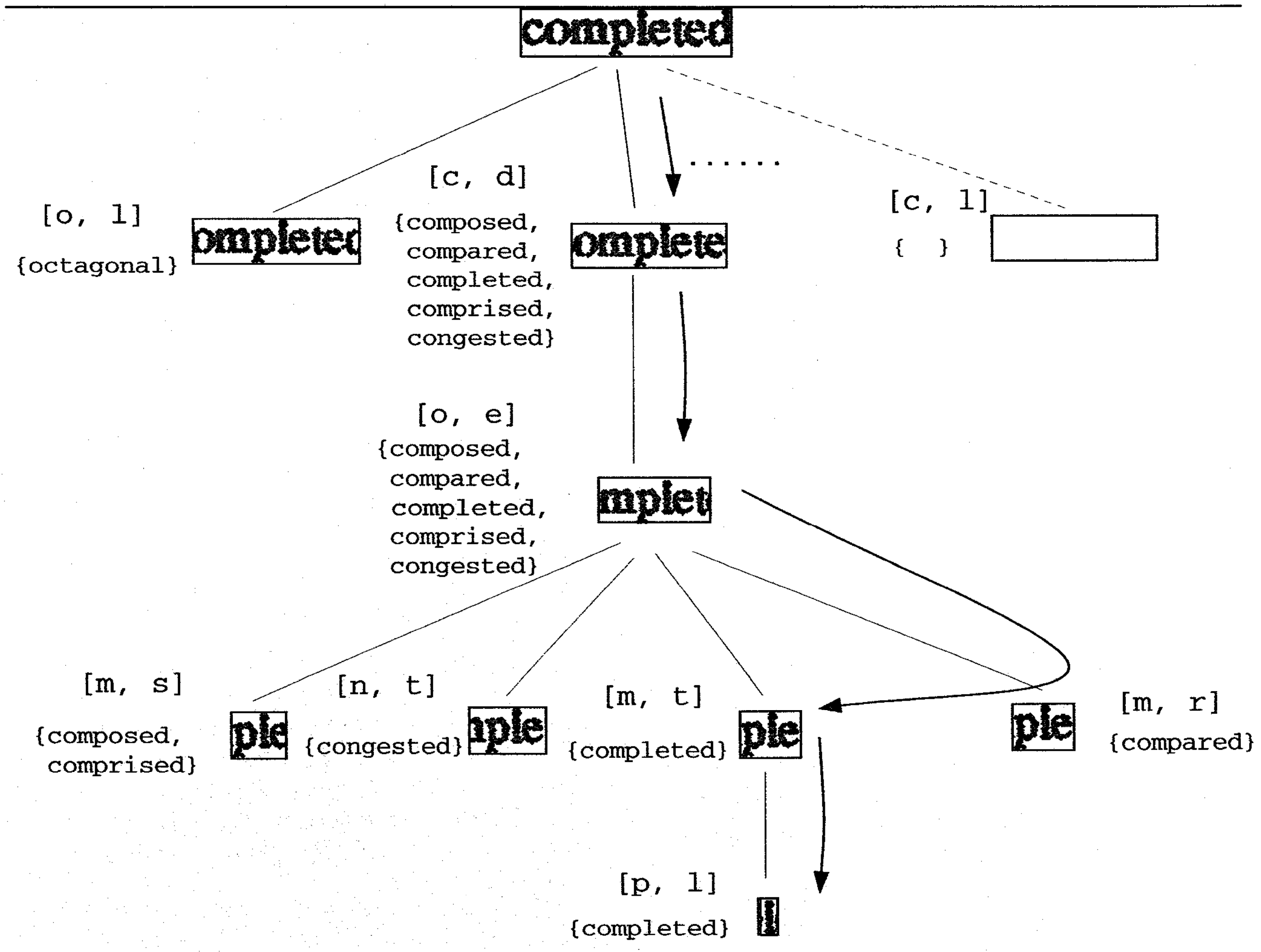


Figure 2: Example application of the hypothesis testing algorithm.

3 Experimental Results

The hypothesis testing algorithm was implemented as part of a word-level deciphering algorithm. This approach first learns the font in which a document is printed from the words it contains that are recognized reliably. The neighborhoods for the remaining words in the document are then input to the A* hypothesis testing algorithm proposed in this paper.

The document image used to test the proposed algorithm consists of two recent news reports on the Korean Peninsula nuclear crisis. These articles contain totally 921 word images and 380 unique words. The document image was generated with the ditroff text formatting package in an 11 pt. Times Roman font. The clean image was printed on plain paper using a laser printer and scanned at 300 ppi resolution. The final binary image is the result of thresholding the scanned gray-scale image by choosing a threshold such that the character distortion and touching characters that are produced test the ability of the algorithm to handle these degradations.

The font information available for the hypothesis testing algorithm which is learned from the recognized portion of text by the deciphering stage includes prototypes for 22 of the 26 lower case characters and a few of the upper case characters.

The proposed hypothesis testing algorithm was able to correctly recognize 370 out of the 374 words left for precise re-recognition, achieving a correct recognition rate of 98.93%. The four cases where the algorithm failed are: "an" being recognized as "au", "21" failed

0-8186-7128-9/95 $10.00 © 1995 IEEE

to come up with any recognition choice because the numerals had not been recognized previously, "Hans" was recognized as "Kaus", and, "sufficient" was recognized as "sediment".

Some interesting observations can be drawn from these errors. In case the word to be recognized is short, such as the word "an", the search algorithm loses its ability to compensate for local mismatches and deformations by relying on global characteristics. This makes the word decisions heavily dependent on local character recognitions that are unfortunately unreliable in presence of image degradations. Short words combined with unknown character prototypes aggravate this problem, as shown in the case of the word "Hans".

4 Conclusions and Future Directions

A hypothesis testing approach to precise recognition of degraded words was presented. Based on partial knowledge of fonts and on word neighborhoods provided by a word recognizer, this approach generates and tests hypotheses for partial word recognitions until a solution of global minimum cost is found. The accumulated cost of reaching a certain partial recognition is combined with a heuristic estimation of potential cost by using an A^* algorithm. The algorithm can recognize degraded words and compensate for local image noise. Tests of the algorithm showed a recognition rate of 98.93% on degraded scanned document images that have many touching characters and character image deformations.

References

[1] R. Casey and G. Nagy, "Recursive segmentation and classification of composite character patterns," In *Proceedings of 6th International Conference on Pattern Recognition*, pages 1023–1026, 1982.

[2] R. Casey and K. Wong, "Document-analysis system and techniques," In R. Kasturi and M. Trivedi, editors, *Image Analysis and Applications*, pages 1–35. New York, 1990.

[3] C. Fang and Jonathan J. Hull, "A Word-level Deciphering Algorithm for Degraded Document Recognition," to appear in *Proceedings of DAIR 95*, Las Vegas, 1995.

[4] P. Hart, N, Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on SSC*, SSC-4:100-107, 1968

[5] T. K. Ho, J. J. Hull and S. N. Srihari, "A Word Shape Analysis Approach to Lexicon Based Word Recognition, *Pattern Recognition letters*, Vol. 13, pp. 821-826, 1992.

[6] J.J. Hull, "Hypothesis generation in a computational model for visual word recognition," *IEEE Expert*, vol. 1, no. 3, pp. 63-70, Fall, 1986.

[7] J.J. Hull, "Hypothesis testing in a computational theory of visual word recognition," *Sixth National Conference on Artificial Intelligence*, Seattle, Washington, pp. 718-722, July 13-17, 1987.

[8] 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.

[9] G. Nagy. "At the frontiers of OCR," *Proceedings of the IEEE*, 80(7), 1992.

[10] S. Peleg and A. Rosenfeld, "Breaking substitution ciphers using a Relaxation Algorithm", *CACM*, vol. 22, no. 11, Nov. 1979, pp. 598-605

[11] J. Schurmann et al, "From pixels to contents," *Proceedings of the IEEE*", vol. 80, no. 7, 1992, pp. 1101-1119

[12] S. Tsujimoto and H. Asada, "Resolving ambiguity in segmenting touching characters," In *Proceedings of the First International Conference on Document Analysis and Recognition(ICDAR-91)*, 1991.

Proceedings of the Third International Conference on Document Analysis and Recognition (ICDAR '95)
0-8186-7128-9/95 $10.00 © 1995 IEEE