Word Recognition With Multi-Level Contextual Knowledge

Tin Kam Ho, Jonathan J. Hull, Sargur N. Srihari

Center for Document Analysis and Recognition
State University of New York at Buffalo
226 Bell Hall
Buffalo, New York 14260, USA

Abstract

A word recognition algorithm is proposed that integrates character recognition with word shape analysis. The algorithm consists of a set of serial filters and parallel classifiers, and the decisions are combined to generate a consensus ranking of the input lexicon. Experimental results with multifont machine-printed word images are discussed.

1 Introduction

A word recognition algorithm is given an image of a word and a lexicon as input and determines the word in the lexicon that best matches the image. Shape information extracted from the image is matched to the shapes of the word prototypes. In addition, knowledge about the context of characters that occur within words in the lexicon is used to improve the matching process. For example, it is well known that the identity of a character is constrained by the identity of its predecessor in the word. There are many such constraints in a fixed lexicon. A number of techniques have been developed to utilize them. Each approach has its particular strengths and is applicable to a subset of the word recognition problem.

There are three major classes of methods for using contextual knowledge in word recognition. These are generally described as character-based, segmentation-based, and word shape analysis. Each of these approaches uses contextual knowledge in a different way. This paper proposes a robust solution to word recognition that uses all three simultaneously. The final decision about a word's identity is made by combining the results of the individual methods.

In character-based word recognition algorithms, the individual characters in a word are first isolated and recognized [3] [14]. The character decisions are then *postprocessed* with a dictionary of allowable words to correct character recognition errors [5]. These methods are suitable for cases where a reliable segmentation can be obtained and the segmented characters are not deformed by normalization. It is also an appropriate strategy for shorter words which are easier to segment and where little word-level context can be utilized. Figure 1 shows some images better recognized by this approach.

Figure 1: Word images better recognized by character based methods.

An alternative to the character based techniques is to defer decisions about character identity and to perform *segmentation-based word recognition.* This is suitable for images where the characters can be easily extracted but are difficult to recognize in isolation. In this approach, feature descriptions of the extracted characters are assembled and matched to a similar representation of the words in a lexicon [7]. This is essentially the approach proposed by McClelland and Rumelhart, where contextual information is integrated in the process

First IAPR International Conference on Document Analysis and Recognition, St. Malo, France, Sept. 30 - Oct. 2, 1991, pp. 905-915.

of character recognition by excitatory and inhibitory links [12]. These techniques effectively focus word-level knowledge on the recognition process and are suitable for situations where characters can be segmented and better recognized together with other characters in the word. Figure 2 shows some images better recognized by this approach.

Figure 2: Word images better recognized by segmentation based methods using word context.

A third type of word recognition algorithm determines features from the whole *word shape* and uses this description to calculate a group of words in a dictionary that match the input [10]. These techniques have been further refined to yield methods that can perform high accuracy recognition [9]. Such methods are especially suitable for images that are difficult to segment into characters, or where the characters are distorted when they are extracted and normalized. Figure 3 shows some example images with these problems that are better recognized by word shape analysis.

Figure 3: Word images better recognized by word shape analysis.

Each of these three methods uses contextual information at a different level. In the character recognition approach, contextual constraints are used only in the last stage, that is, the postprocessing stage after the character classes are decided. In the segmentation-based word recognition approach, contextual information is used before the class decisions are made, but after feature extraction. In the word shape analysis approach, word context information is used directly in feature computation and matching.

Most previous solutions to word recognition have used only one of the three methods outlined above. Since each of them excels for a limited type of image, their performance and generality have been restricted. A particular method may effectively recognize text in a given, well-specified domain, but might fail miserably when presented with another type of text. To achieve the goal of complete text recognition, a methodology is needed that effectively employs all the techniques listed above. The rest of this paper proposes a robust word recognition algorithm that integrates all these methods and simultaneously uses word context at different levels. Reliability is achieved by making use of multiple classifiers with uncorrelated errors at various stages. Final decisions are derived by applying a group consensus function that measures agreement among the parallel classifiers.

2 A Word Recognition Algorithm

The proposed word recognition algorithm consists of a set of filters, a collection of classifiers, and a decision combination mechanism. The filters are applied to the input lexicon to reduce the set of classes to be discriminated. At the end of the sequential filtering stages, there is a layer of parallel classifiers. These classifiers produce independent rankings of the filtered lexicon. The rankings are then combined by a decision combination mechanism to produce a final ranking, which is the output of the algorithm.

The parallel classifiers are based on three different approaches, which are (1) character recognition methods, (2) segmentation-based word recognition, and (3) word shape analysis methods. Contextual knowledge of different levels is used in the three approaches.

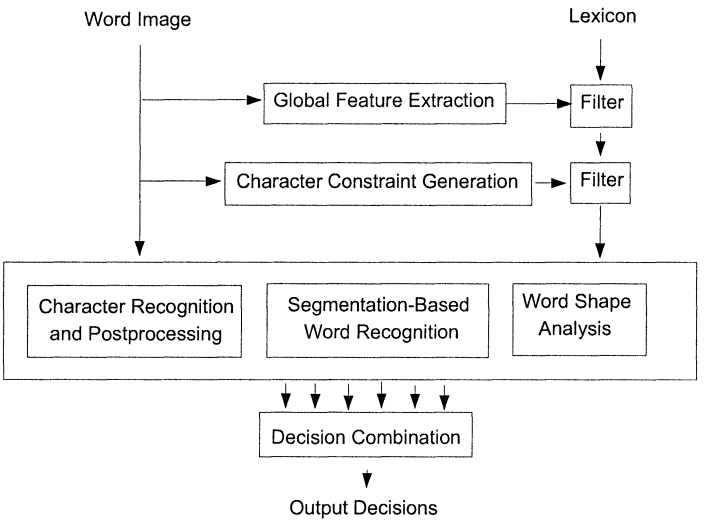

Figure 4: The proposed word recognition algorithm.

A decision combination strategy combines the rankings produced by the collection of classifiers and derives a consensus ranking. Confidence scores for each word in the lexicon are computed using the rankings. A consensus ranking is derived by sorting the words by the confidence scores.

The following sections describe details of an implementation of each component in the proposed word recognition algorithm and experimental results on real world data.

3 Lexicon Filtering by Global Features

Global features are the wholistic and simpler aspects of the word shape that can be easily and reliably measured. They are used to filter the input lexicon to simplify subsequent classification tasks. The global features that are useful for this purpose include estimates of the word length and the word case. If the estimates are accurate, only the words with lengths matching the estimated length need to be retained in the lexicon, and the words can then be converted to the estimated case.

It is difficult to estimate word length precisely, therefore an interval estimate is attempted instead. This is done by first performing character segmentation, and then relaxing the character count into an interval by examining the variations in sizes of the segmented characters.

Word case can be estimated by examining variations in the sizes and alignment of the connected components. The result is confirmed by an analysis of the heights from the located top line and base line to the upper and lower boundaries of the image. If no agreement is obtained, the word case will be left undetermined.

4 Lexicon Filtering by Reliable Character Constraints

The degradation across a set of images from the same source is not always uniform. The amount of degradation can vary from word to word, or within the same word. When the words are partially degraded, there may still be some visual cues that can be easily and reliably identified. It is advantageous to utilize the maximum amount of reliable information from the image as early as possible to reduce the set of classes.

Reliability is the most important concern in the lexicon filtering stages, since this is a sequential process

and any words filtered out cannot be recovered. One useful way to achieve reliability is to obtain the agreement of several independent decisions. This is the approach we take at this stage to derive reliable information about the characters in the word.

The lexicon can be much constrained if any of the characters in the word can be correctly recognized. This requires correct character segmentation and recognition for all kinds of input images. It is difficult to obtain perfect recognizers. Nonetheless, reliability can be achieved by using several satisfactory recognizers with uncorrelated errors.

Without loss of generality, we can assume that each of the recognizers ranks the true character classes by their similarity to the input character. If all the classifiers agree at the top decision, that decision is believed to be reliable, with a few exceptions that decisions like I's or l's could be splits. A few heuristic rules can be written to ignore these decisions. Using the reliable decisions, a set of constraints can be constructed. Since we are looking for exact matches, the constraints can be conveniently represented by regular expressions. The constraints can be constructed in such a way that precise positioning of the character in the word is not required, and thus can tolerate some segmentation errors. The words can then be graded by the number of constraints they match. A ranking of the words is produced by this grading. A threshold can be applied on the grades to filter out the unlikely words. Figure 5 shows an example word image, the character decisions and the constraints.

```
Decisions by 4 classifiers:
1 GOu A4R KnR L1t A34 NMa JD0
2 GWS AaR aMN Lt6 AaR NHU DU0
3 GWB A4n rXx LaA A4i nNH aOo
4 GSE AaS KXE LrI A86 NMW 2B8

reliable decisions: G A ? L A ? ?

character constraints:
1 (.?)[G].....(.?|..)
2 (.?|..)[A]....(.?|..)
3 ..(.?|..)[L]..(.?|..)
4 ...(.?|..)[A].(.?|..)
5 [G][A].[L][A]..
```

```
where
. matches any single character
.? matches zero or one character
(.?|..) matches zero, one, or two characters
```

Figure 5: An example image, the character decisions and derived constraints.

For images that are of very bad quality or incorrectly segmented, there may be no reliable decisions at all so no constraints can be established. In those cases all words input to this filter will pass.

5 Lexicon Ordering by Character Recognition and Postprocessing

The character recognition approach illustrated in Figure 6 makes use of a character segmentation algorithm, which separates the word image into individual character images. A character recognition algorithm is then applied to identify the individual images. The character decisions are then post-processed against a lexicon. Contextual knowledge is used only at this latest stage. A ranking of the lexicon can be generated by the postprocessing algorithm.

A fuzzy template matcher is used to perform character recognition. This recognizer makes use of neighboring pixel values in computing a distance between corresponding pixels in the input and stored templates.

A heuristic string matching algorithm postprocesses the decisions versus the lexicon. The algorithm takes two decisions for each character, and constructs a set of strings using all top decisions and by replacing each top

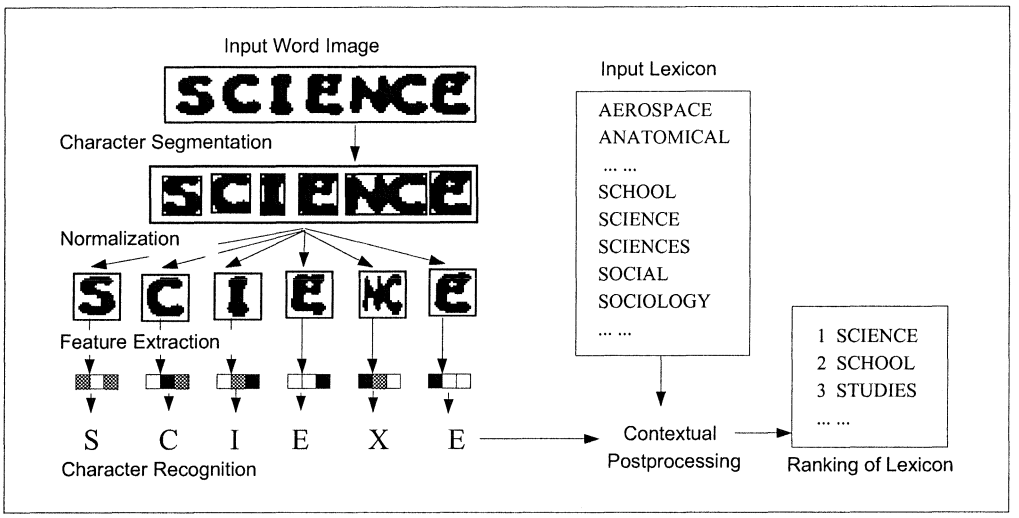

Figure 6: The Character Recognition Approach.

decision in turn with the second choice. The strings are then matched with the lexicon words of the same lengths, and those with one character longer or shorter. The matches are graded by the number of matching characters. A penalty is added to the matches that correspond to second choices. In matching the unequal length words, a character is removed from each position of the longer string in turn. A penalty is added to degrade the matches of unequal length words. The overall score for a word is the highest of its scores from all possible matches. The lexicon is then ranked by the scores.

When weight scores are associated with the constraints in the regular expression matcher used in character constraint filtering, a ranking of the lexicon can also be generated. A high score is associated with a constraint given by a reliable character decision. This is another form of contextual postprocessing that allows some fuzziness in character positions.

6 Lexicon Ordering by Segmentation-Based Word Recognition

This approach uses an intermediate level of context between the character recognition approach and the word shape method (Figure 7). A word image is first segmented into individual character images. Features are extracted from the segmented character images and represented by feature vectors. These character feature vectors are then concatenated and matched with similar feature vectors for the lexicon words. Hence the character features are compared under the contextual constraints represented by a lexicon.

The features can be as simple as pixel values. Each segmented character is normalized to a 24x24 grid. The pixel values of each segmented character are then concatenated to form a word feature vector. Thus a word segmented into 4 characters has a feature vector of 24x24x4 = 2304 elements.

In the matching process, a distance measure is computed for word feature vectors of the same length. To allow for segmentation errors, lexicon words with lengths one more or one less than the input image are also matched. This is done by deleting one character at a time from different positions of the longer vector. The distance is the number of different elements divided by the length of the words compared. Since character segmentation is correct in many cases, a weight is used to decrease the distance scores for equal length vectors and increase the scores for unequal length vectors. A ranking of the lexicon is then produced by sorting the words in order of increasing distance.

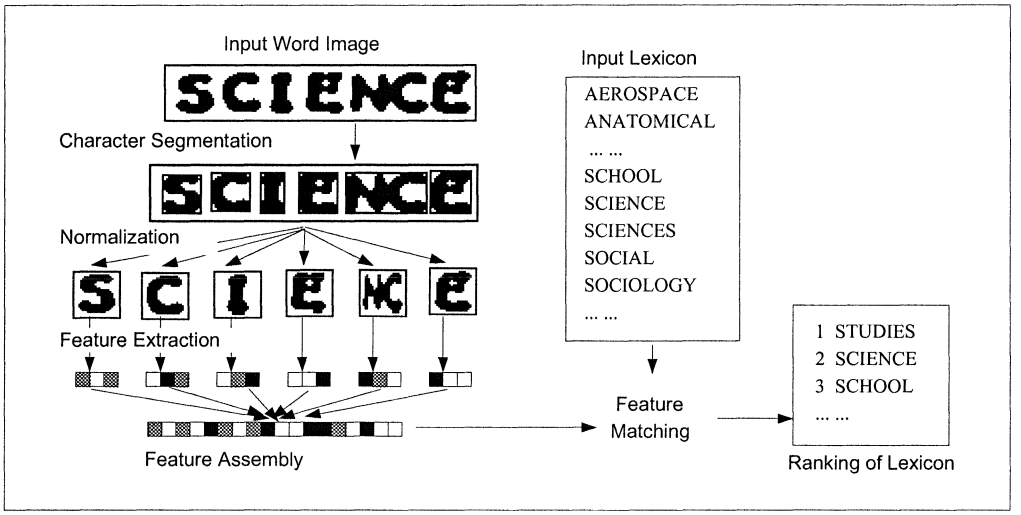

Figure 7: The Segmentation-Based Word Recognition Approach.

7 Lexicon Ordering by Word Shape Analysis

This approach attempts to describe and compare the shape of the word as a whole object (Figure 8). Features that describe the details of the word shape are extracted and their relative locations are recorded in a feature vector. The feature vector is matched to a lexicon of words and a ranking of the lexicon is produced.

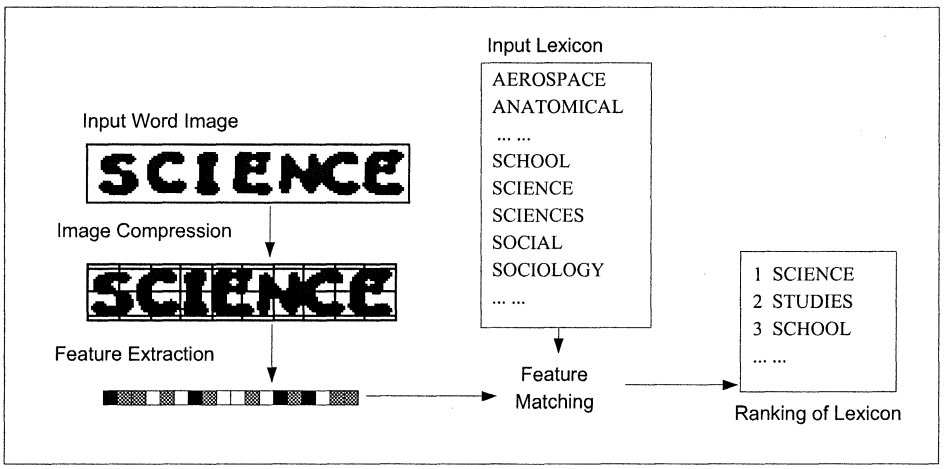

Figure 8: The Word Shape Analysis Approach.

The shape features are from two sets: (1) template defined features and (2) stroke direction distribution features. Each of these feature sets provides different information about the shape of a word. We use descriptors that register the feature locations with reference to a global frame. This global frame consists of the four reference lines that divide the vertical axis into the ascender region, the middle region, and the descender region, and ten equal-sized divisions along the horizontal axis. The middle vertical region is further divided into upper and lower parts. As a result, the image area is partitioned into 4 vertical regions, and 10 horizontal regions, i.e., 40 cells.

Figure 9 shows the area partitions given by such a frame.

Figure 9: An image and its 40 area partitions.

Template Defined Features This feature set is defined by 32 7x7 templates defined in [2]. A word image is first normalized to 24 pixels in cap-height (the height between the upper boundary and the base line). It is then convolved with the 32 templates and responses are thresholded. Each nonzero response represents that a feature of a particular type is detected at that pixel position. The outputs are described by a 1280-dimensional feature vector which stores counts of the 32 features detected in the 40 cells, normalized by the sum of counts over the image.

Stroke Direction Distribution The stroke direction distribution captures the spatial distribution of black pixels across the image. Each black pixel is first labeled as belonging to a stroke of one of four different directions. This labeling is done using the *local direction contribution* method suggested in [13]. At each black pixel in the image, the length of the current run in each of the four directions east-west, northeast-southwest, north-south, and northwest-southeast is computed. The pixel is labeled with the direction in which the *run length is a maximum.* That is, each black pixel is labeled as part of a stroke of one of the four directions. Figure 10(a)-(e) shows an example of such pixel labeling. The distribution is given by the count of labeled black pixels of each type at every area partition in the image. The stroke direction distribution is described by a 160-dimensional feature vector, which stores counts of black pixels of each of the four types in the 40 cells.

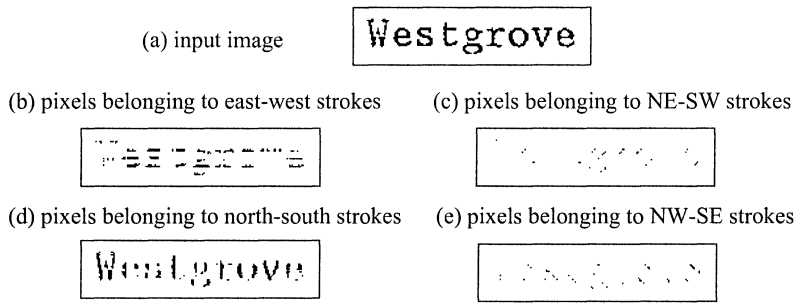

Figure 10: Example of computing the stroke direction distribution.

The word shape recognition approach uses the contextual knowledge that is contained in a lexicon directly in the process of feature matching. The two feature vectors that describe the word shape are matched with similar feature vectors for the words in the lexicon. The matching is performed with a nearest-neighbor classifier that uses a city block distance metric to compare a feature vector for an input word to a feature vector for a dictionary word [6].

The feature vectors for dictionary words are derived by first creating a prototype word image by appending the images of characters from a font sample. The feature vectors are then calculated from this image. A number of fonts are used to cover variations in character shapes to guarantee good performance. Two rankings are produced in this stage, one using the template defined features and the other using the stroke direction distribution features.

8 Decision Combination

The combination algorithm uses the results of the selected classifiers to generate a consensus decision [8]. As each independent classifier outputs a ranking of the lexicon, a confidence score is computed using these rankings. A consensus ranking is then generated by sorting the words by the computed confidence scores.

Three different confidence functions are proposed for this purpose. The first one is a highest rank method. The score for each word is the highest rank among the ranks it receives from all the classifiers. Therefore, a word receives a high score as long as there is one classifier that ranks it highly. The combined ranking is given by sorting the words by the highest ranks. This method is particularly useful in reducing a large lexicon to a small neighborhood, since it takes advantage of the best classifier for each input case.

The second function is called the Borda count [4]. For a set of rankings on the same set of classes, the Borda count for each class is the sum of the numbers of classes ranked below it by each classifier. The combined ranking is given by arranging the classes in descending Borda count. Intuitively, if a class is ranked near the top by more classifiers, its Borda count tends to be larger and will be closer to the top in the combined ranking. It is a measure of agreement among the classifiers.

The third method generalizes the Borda count by adding weights to each ranking. The confidence score for each word is computed as a weighted sum of the ranks it receives from the classifiers. The weight for each classifier can be estimated using a special form of regression analysis that involves a logit transformation [1]. Using a training set of decisions, the significance of contribution of each classifier can be statistically tested.

The lexicon is first reduced by the highest rank method. The ranks of the words in the neighborhood are then combined using the Borda count and the estimated logistic regression function.

It is not necessary that all the parallel classifiers are applied to each input image. Dynamic selection of appropriate classifiers is possible, if an effective measure of the image quality can be defined. The algorithm can also dynamically select a single or a combination of classifiers based on the quality measure.

9 Experimental Results

The recognition system has been developed using a collection of images of *machine-printed* postal words obtained from live mail. They were scanned on a postal optical character reader at roughly 200 pixels per inch and binarized. The font and quality of the images vary. Figure 11 shows some example images in our data set. As can be seen in these examples, the image quality is highly variable and the characters can be severely broken or touching each other. The unrestricted font variations also can cause substantial failures in a conventional text recognition system.

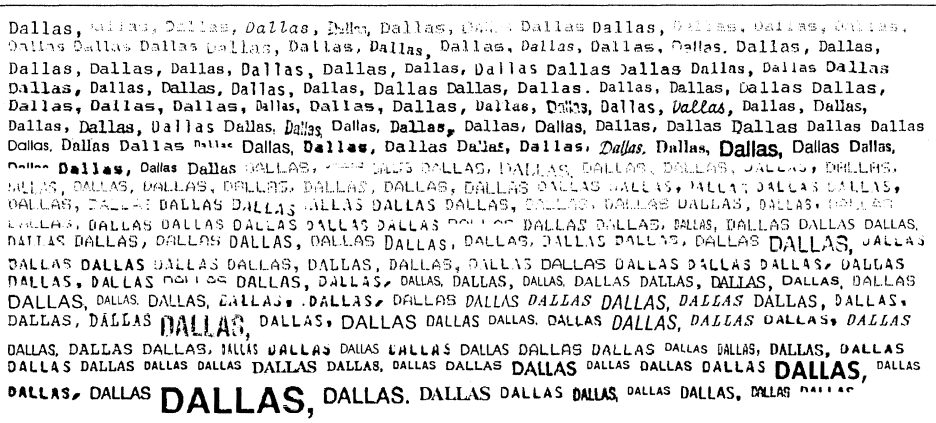

Figure 11: Examples of image degradation and font style variations included in the test set.

In the experimental system, we used five different word classifiers in total. These include (1) a fuzzy character template matcher with a heuristic contextual postprocessing algorithm, (2) six character recognizers with decisions postprocessed by a regular expression matcher, (3) a segmentation based word recognizer with pixel values as features, (4) a word shape analyzer using stroke direction features, and (5) a word shape analyzer using Baird template features. The six character recognizers include a pixel based NNC (nearest neighbor classifier) (i.e., a template matcher) and a Bayesian classifier, a Baird feature based NNC and a Bayesian classifier, a stroke direction feature based NNC and a fuzzy pixel value based NNC (the fuzzy template matcher). No dynamic selection was applied.

The feature extractors were developed and modified by observing performance on a small initial training set of about 200 images. The character recognizers were trained on a set of 19151 sample characters extracted from 400 address blocks. The prototypes of stroke direction vectors were produced by averaging those from 77 font samples. For Baird feature vectors, 57 fonts were used. A set of 76 fonts was used to construct prototypes for the segmentation based method.

A set of 1055 word images was used to generate decisions from the word classifiers. A lexicon of 33850 words was used that included all the true words in the images. Each word in the lexicon is represented in both upper and mixed cases. In the application domain, purely lower case is not used. The results of this run were used to estimate the regression parameters for decision combination. All the five classifiers, as well as their combination by highest ranks, were determined to be statistically significant. The estimated weights were 0.0539, 0.1880, 0.2098, 0.3292 and 0.1775 for the five classifiers respectively, and 0.1864 for the ranking given by the highest rank method.

Another set of 1671 images was used to test the overall algorithm. The same lexicon of 33850 words was used. Table 1 summarizes the performance of the five individual classifiers and results at several stages of decision combination. Three subsets of sizes 10000, 5000, and 1000 were also selected from the input lexicon. The objective was to determine the potential effect of a global contextual knowledge source that could provide reduced lexicons to word recognition thereby improving its performance. Each of the subsets of the original lexicon contains all the true words in the images. The rest of the words in the subsets were randomly selected.

Table 1: Summary of Performance on 1671 Test Images Using a 33850 Word Lexicon
(% Correct at Top N Decisions)

Descriptions	1	2	3	10	50	100	500
1) word shape with stroke direction features	42.4	53.7	59.5	72.1	81.9	84.9	90.8
2) word shape with Baird features	58.9	70.0	74.5	82.9	88.5	90.2	93.2
3) char recog with regular expression matcher	76.9	83.2	85.4	88.3	91.8	93.2	95.0
4) char recog with heuristic postprocessor	79.2	86.1	88.2	90.5	92.7	93.5	94.8
5) segmentation based word recognition	74.8	84.1	86.3	90.5	93.4	94.6	95.5
6) combination of 1)-5) by highest rank method	46.6	73.8	87.0	94.9	97.3	97.6	98.6
7) combination of 1)-6) by Borda count	83.1	88.1	90.7	94.6	96.3	97.0	98.6
8) combination of 1)-6) by logistic regression	88.4	91.2	92.7	95.1	97.4	98.1	98.6
9) results of 8) with case (upper,mixed) merging	88.7	91.4	92.9	95.2	97.6	98.5	98.9
10) results of 9) with word length filtering	88.9	91.6	92.9	95.3	97.8	98.4	98.7
11) results of 10) in a 10000 word subset	92.6	94.0	94.9	97.0	98.6	98.8	98.9
12) results of 10) in a 5000 word subset	94.1	95.0	96.0	97.8	98.7	98.9	98.9
13) results of 10) in a 1000 word subset	95.5	97.1	97.9	98.7	98.9	98.9	98.9

Using the highest rank method of decision combination, the lexicon was reduced to a neighborhood of 500 words with a 98.6% accuracy, which was not achieved by any of the individual classifiers. This method did not give a high correct rate at the top choice because of ties among the top 5 decisions. Better consensus ranking was achieved by the Borda count and the logistic regression method. There was a 9.2% gap between the top

choice correct rate of the combination by logistic regression (8) and that of the best individual classifier (4). The filtering stages were useful in reducing run time but had no great impact on the rankings. The performance on the random subsets shows that other higher level contextual constraints on the lexicon can improve the final rankings significantly.

10 Conclusions

A robust word recognition algorithm has been developed that uses three approaches that utilize contextual knowledge at different levels. These include a character recognition approach, a segmentation based approach, and a word shape analysis approach. A control strategy is designed to combines their decisions.

In an experiment with 1671 word images and a 33850 word lexicon, the proposed algorithm achieved a correct rate of 88.9% at top choice and 95.3% in the top 10 choices. When the input lexicon is reduced to 1000 words, a correct rate of 95.5% at top choice and 98.7% in the top 10 choices was achieved. The performance of the algorithm is significantly better than each of the individual classifiers applied in isolation. Future work includes further refinement of the control and decision combination strategies towards more flexible dynamic adaptation to both top-down and bottom-up constraints.

Acknowledgements The support of the Office of Advanced Technology of the United States Postal Service is gratefully acknowledged. Jiah-Shing Chen provided the fuzzy template matcher for character recognition. Yan Li and Liang Li helped in implementing the segmentation based word recognition algorithm. Dr. T.S. Lau provided helpful comments. Peter Cullen, Michal Prussak, Piotr Prussak and Ralph Ames assisted in the development of the database for the experiments. The authors highly appreciate their help.

References

[1] A. Agresti, *Categorical Data Analysis*, John Wiley & Sons, 1990.

[2] H.S. Baird, H.P. Graf, L.D. Jackel, W.E. Hubbard, A VLSI Architecture For Binary Image Classification, in *From Pixels to Features* , J.C. Simon (editor), North Holland, 1989, 275-286.

[3] H.S. Baird, K. Thompson, Reading Chess, *IEEE Transaction of Pattern Analysis and Machine Intelligence*, **PAMI-12**, 6, June 1990, 552-559.

[4] D. Black, *The Theory of Committees and Elections*, Cambridge University Press, London, 1958, reprinted 1963.

[5] A.C. Downton, E. Kabir and D. Guillevic, Syntactic and Contextual Post-Processing of Handwritten Addresses for Optical Character Recognition, *Proceedings of the 9th International Conference on Pattern Recognition*, 1988, 1072-1076.

[6] R.O. Duda, P.E. Hart, *Pattern Classification And Scene Analysis*, Addison-Wesley, New York, 1973.

[7] R.A. Duderstadt, M.J. Cykana, A.V. Monfared, and D.H. Gibbs, Isolated Word Recognition for Postal Address Processing, *Proceedings of the 4th USPS Advanced Technology Conference*, November 1990, 233-245.

[8] T.K. Ho, J.J. Hull, S.N. Srihari, Combination of Structural Classifiers, *Pre-Proceedings of the IAPR Syntactic and Structural Pattern Recognition Workshop*, New Jersey, June 13-15, 1990, 123-136.

[9] T.K. Ho, J.J. Hull, S.N. Srihari, A Word Shape Analysis Approach to Recognition of Degraded Word Images, *Proceedings of USPS Advanced Technology Conference*, November 1990, 217-231.

[10] J.J. Hull, Hypothesis Testing in a Computational Theory of Visual Word Recognition, *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI)*, Seattle, Washington, July 13-17, 1987, 718-722.

[11] D.S. Lee, S.W. Lam, S.N. Srihari, A Structural Approach to Recognize Hand-printed and Degraded Machine-printed Characters, *Pre-Proceedings of the IAPR Syntactic and Structural Pattern Recognition Workshop*, New Jersey, June 13-15, 1990, 256-272.

[12] J.L. McClelland, D.E. Rumelhart, An Interactive Activation Model of Context Effects in Letter Perception: Part 1. An Account of the Basic Findings, *Psychological Review*, **88**, 5, September, 1981, 375-407.

[13] S. Mori, K. Yamamoto, M. Yasuda, Research on Machine Recognition of Handprinted Characters, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-6**, 4, July 1984, 386-405.

[14] C.Y. Suen, C. Nadal, T.A. Mai, R. Legault, and L. Lam, Recognition of Totally Unconstrained Handwritten Numerals Based on the Concept of Multiple Experts, *First International Workshop on Frontiers in Handwriting Recognition*, Concordia University, Montreal, Canada, April 2-3, 1990, 131-143.