ON-LINE CURSIVE HANDWRITING RECOGNITION USING SPEECH RECOGNITION METHODS

Thad Starner†, John Makhoul, Richard Schwartz, and George Chou

BBN Systems and Technologies
70 Fawcett Street, Cambridge, MA 02138
Email: Makhoul@bbn.com

ABSTRACT

A hidden Markov model (HMM) based continuous speech recognition system is applied to on-line cursive handwriting recognition. The base system is unmodified except for using handwriting feature vectors instead of speech. Due to inherent properties of HMMs, segmentation of the handwritten script sentences is unnecessary. A 1.1% word error rate is achieved for a 3050 word lexicon, 52 character, writer-dependent task and 3%-5% word error rates are obtained for six different writers in a 25,595 word lexicon, 86 character, writer-dependent task. Similarities and differences between the continuous speech and on-line cursive handwriting recognition tasks are explored; the handwriting database collected over the past year is described; and specific implementation details of the handwriting system are discussed.

1. INTRODUCTION

Traditionally, the first step in handwriting recognition is the segmentation of words into component characters [1]. However, in modern continuous speech recognition efforts, phonemes are not segmented before training or recognition. Instead, segmentation occurs simultaneously with recognition. If such a system could be adapted for handwriting, the very difficult and time consuming issue of segmentation could be avoided. This paper presents an approach for the automatic recognition of *on-line cursive handwriting* (using input from a pentop computer) by using continuous speech recognition methods. Specifically, the use of hidden Markov models and statistical grammars is explored. We show that, with essentially no modification, a speech recognition system can perform accurate on-line handwriting recognition, with the input features being those of the writing instead of speech.

Hidden Markov models have intrinsic properties which make them very attractive for handwriting recognition. For training, all that is necessary is a data stream and its transcription (the text matching the handwriting). The training process automatically aligns the components of the transcription to the data. Thus, no special effort is needed to label training data. Segmentation, in the traditional sense, is avoided altogether. Recognition is performed on another data stream. Again, no explicit segmentation is necessary. The segmentation of words into characters or even sentences into words occurs naturally by incorporating the use of a lexicon and a language model into the recognition process. The result is a text stream that can be compared to a reference text for error calculation.

Section 2 discusses the similarities of speech and handwriting recognition tasks and provides some background on techniques. Section 3 describes an initial 3050 word, 52 symbol, writer dependent experiment. Section 4 discusses a more ambitious 25,595 word, 86 symbol, writer dependent system involving multiple writers. Section 5 examines the results of this experiment and discusses future work.

2. SIMILARITIES OF ON-LINE HANDWRITING RECOGNITION TO SPEECH RECOGNITION

On-line handwriting is very similar to the problem of continuous speech recognition. On-line handwriting can be viewed as a signal (x,y coordinates) over time, just like in speech. The items to be recognized are well-defined (usually the alphanumeric characters) and finite in number, as are the phonemes in speech. The shape of a handwritten character depends on its neighbors. Spoken phonemes also change due to coarticulation in speech. In both cases, these basic units form words and the words form phrases. Thus, language modeling can be applied to improve recognition performance.

In spite of these similarities, handwriting recognition has some basic differences to speech recognition. Unlike continuous speech, word boundaries are usually distinct in handwriting. Thus, words should be easier to distinguish. However, in cursive writing the dots and crosses involved in the characters "i", "j", "x", and "t" are not added until after the whole word is written. Thus, all the evidence for a character may not be contiguous. Additionally, in words with multiple crossings ("t" and "x") and/or dottings ("i" and "j") the order of pen strokes is ambiguous. Even so, with the many parallels between on-line writing and speech, speech recognition methods should be applicable to on-line handwriting recognition. Since hidden Markov models currently constitute the state of the art in speech recognition, this method also seems a likely candidate for handwriting recognition.

There has been some interest in the use of HMMs for on-line handwriting recognition (see, for example, [2,3]). How-

†currently with the MIT Media Lab.

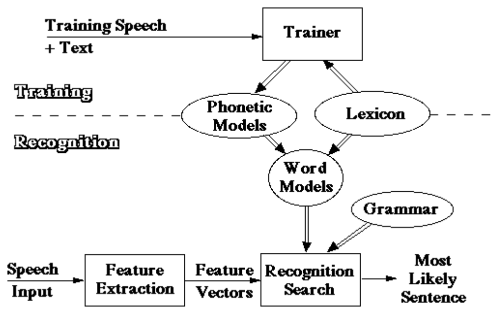

Figure 1: BYBLOS speech system.

Figure 2: Connecting strokes.

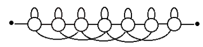

Figure 3: 7-state HMM used to model each character.

ever, the few studies that have used HMMs have dealt with small vocabularies, isolated characters, or isolated words. In this study, our objective is to deal with continuous cursive handwriting and large vocabularies (thousands of words), using a speech recognition system and language models.

3. INITIAL SYSTEM

In the initial system, the BBN BYBLOS Continuous Speech Recognition system [4] (see Figure 1) was used without modification on an on-line cursive handwriting corpus created from prompts from the ARPA Airline Travel Information Service (ATIS) corpus [5]. These full sentence prompts (approximately 10 words per sentence) were written by a single subject. These sentences were then reviewed (verified) to make sure that the prompts were transcribed correctly. After verification, these sentences were separated into a set of 381 training sentences and a mutually exclusive set of 94 test sentences. The lexicon for this task included 3050 words, where lowercase and capitalized versions of a word are considered distinct.

For this initial system there were 54 characters: 52 lower and upper case alphabetic, a space character, and a "backspace" character. The backspace character is appended onto words that contain "i", "j", "x", or "t". This character models the space the pen moves after finishing the body of the word to add the dot or the cross when drawing one of these characters.

The data was acquired using a Momenta pentop which stored the script in a simple time series of x and y coordinates at a sampling rate of 66 Hz. The handwriting data is sampled continuously in time, except when the pen is lifted (Momenta pentops provide no information about pen movement between strokes). Because we wanted to use our speech recognition system with no modification, we decided to simulate a continuous-time feature vector by arbitrarily connecting the samples from pen-up to pen-down with a straight line and then sampling that line ten times. Thus, the data effectively became one long criss-crossing stroke for the entire sentence, where words run together and "i" and "j" dots and "t" and "x" crosses cause backtracing over previously drawn script (see Figure 2).

For each sample point, an analysis program computed a two-element feature vector: the writing angle at that sample and the change in the writing angle [2]. These time series of feature vectors were then fed into the BYBLOS system. For this task, BYBLOS quantizes the feature vectors for a sentence into 64 different clusters. These new time series are then used with their respective sentence transcriptions to train HMMs representing the script characters (note that the alignment of the clusters with the sentence transcriptions occurs automatically in this process). A 7-state HMM model was chosen to represent each symbol (see Figure 3). Since the penning of a script letter often differs depending on the letters written before and after it, additional HMMs are used to model these contextual effects [5]. Adjacent effects between two letters (bilets) are modeled as well as three letter (trilet) contexts. In a given set of sentences there may be many trilets, up to the number of symbols cubed. However, in English only a subset of these are allowed. In the ATIS task there are 3639 different trilets in the training sentences.

A statistical grammar can also be used to improve recognition performance. For this experiment, a bigram grammar (to relate pairs of words) was created using a larger set of 17209 sentences from the ATIS corpus (the 94 test sentences were not included). The resultant grammar has a perplexity of 20. Table 1 shows the word error rates for this task when doing recognition using context without the grammar (perplexity = 3050), using the grammar without context, and using both context and the grammar. Word error rate is measured as the sum of the percentage of words deleted, the percentage of words inserted, and the percentage of words that are substituted for other words in the set of test sentences.

Table 1: ATIS 3050 word, writer-dependent test results

	context + no gram.	*no context + gram.*	*context + gram.*
word error rate	4.2%	2.2%	1.1%

As can be seen from the table, both context and a grammar are very powerful tools in aiding recognition. With no grammar but with context an error rate of 4.2% was observed. When the grammar was added and context not used, the error rate dropped to 2.2%. However, the best result used both context and a grammar for an word error rate of 1.1%. Of interest is the factors of 2 relating the error rates shown. Similar factors of 2 have also been observed

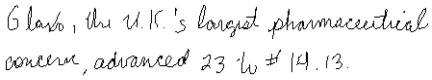

Figure 4: Writing from subject aim.

in the research on the speech version of this corpus. With the best (1.1%) word error rate, only 10 errors occurred for the entire test set. Experimentation was suspended at this point since so few errors did not allow any further analysis of the problems in our methods.

The above experiments demonstrated the potential utility of speech recognition methods, especially the use of HMMs and grammars, to the problem of on-line cursive handwriting recognition. Based on these good preliminary results, we embarked on a more ambitious task with a larger vocabulary and more writers.

4. WALL STREET JOURNAL: A 25,000 WORD TASK

Recently, we have collected cursive written data using text from the ARPA Wall Street Journal task (WSJ) [6], including numerals, punctuation, and other symbols, for a total of 88 symbols (62 alphanumeric, 24 punctuation and special symbols, space, and backspace). The prompts from the Wall Street Journal consist mainly of full sentences with scattered article headings and stock listings (all are referred to as sentences for convenience). We have thus far collected over 7000 sentences (175,000 words total or about 25 words/sentence) from 21 writers on two GRiD Convertible pentops. See Figure 4 for an example of the data collected (this sentence was taken from a test set). The writers were gathered from the Cambridge, Massachusetts area and were mainly students and young professionals. Several non-native writers were included (writers whose first working language was not English). While the handwriting input was constrained, the rules given the subjects were simple: write the given sentence in cursive; keep the body of a word connected (do not lift the pen in the middle of a word); and do crossings and dottings after completing the body of a word. However, since many writers could not remember how to write capital letters in cursive, great leniency was allowed. Furthermore, apostrophes were allowed to be written both in the body of the word, or at the end of the word like a cross or dot. For example, the word "don't" could be written as "dont" followed by the placement of the apostrophe or "don", apostrophe, and "t". Overall, this task might be best described as "pure cursive" in the handwriting recognition literature.

For the purposes of this experiment, punctuation, numerals, and symbols are counted as words. Thus, ".", ",", "0", "1", "$", "{", etc., are each counted as a word. However, apostrophes within words are counted as part of that word. Again, a capitalized version of a word is counted as distinct from the lowercase version of the word. While these standards may artifically inflate the word error rates, they are a simple way to disambiguate the definition of a word.

In addition to the angle and delta angle features described in the last section, the following features were added: delta x, delta y, pen up/pen down, and sgn(x - max(x)). Pen up/pen down is 1 only during the ten samples connecting one pen stroke to another; everywhere else it is 0. Sgn(x - max(x)) is 1 only when, at that time, the current sample is the right-most sample of the data to date. Also, two preprocessing steps were used on the subjects' data. The first was a simple noise filter which required that the pen traverse over one hundredth of an inch before allowing a new sample. The second step padded each pen stroke to a minimum size of ten samples.

At the time of this writing, samples from six subjects were used for writer dependent experiments. Three fourths of a subject's sentences were used for training with the remaining fourth used for testing (see Table 2). A lexicon of 25,595 words was used since it spanned all of the data. A bigram grammar was created from approximately two million Wall Street Journal sentences from 1987 to 1989 (not including the sentences used in data collection). The results of the writer dependent tests are shown in Table 3. Substitution, deletion, insertion, and the total word error rates are included. Table 4 shows estimated character recognition error rates for each class of character: alphabetic, numeral, and punctuation and other symbols. The sum of the substituion and deletion error rates for each class is represented in this table since insertions are not directly attributeable to a particular class of character. However, the total character error shown incorporates insertion errors since these errors are distributed over the entire set of classes. On average, the test sets consist of 1.9% numerals, 4.1% punctuation and other symbols, and 94% alphabetics. Both aim and shs are non-native writers. A test experiment was performed without a grammar (but with context) on subject shs resulting in an error rate approximately four times the previous error rate. This result was the same ratio seen in the ATIS task.

Table 2: Division of subjects' sentences into training and test

subject	*# train sentences*	*# test sentences*
aim	423	141
dsf	404	135
rgb	437	146
shs	423	141
slb	411	137
wcd	314	105

Table 3: WSJ 25,595 word, writer dependent word errors

subject	*Subst.*	*Delet.*	*Insert.*	*Total*
aim	2.7%	0.4%	1.4%	4.5%
dsf	3.6%	0.4%	1.2%	5.2%
rgb	3.3%	0.5%	1.7%	5.5%
shs	2.0%	0.2%	0.7%	2.9%
slb	2.9%	0.1%	1.3%	4.3%
wcd	2.1%	0.4%	0.5%	3.0%
ave.	2.8%	0.3%	1.1%	4.2%

Table 4: Estimated character error rates for alphabetics, numerals, and symbols

subject	*Est. num.*	*Est. sym.*	*Est. alpha.*	*total*
aim	7.1%	4.7%	.47%	1.4%
dsf	8.3%	8.6%	.78%	1.9%
rgb	3.2%	11.%	.77%	1.8%
shs	4.6%	7.8%	.26%	0.80%
slb	7.2%	7.1%	.64%	1.7%
wcd	5.4%	5.7%	.47%	1.0%
ave.	6.2%	7.5%	.57%	1.4%

5. EXPERIMENT ANALYSIS AND FUTURE DIRECTIONS

These results are quite startling when put in context. The BYBLOS speech system was not significantly modified for handwriting recognition, yet it handled several difficult handwriting tasks. Futhermore, none of the BYBLOS automatic optimization features were used to improve the results of any writer (or group of writers). No particular stroke order was enforced on the writers for dottings and crossings (besides being after the body of the word), and there are known inaccuracies in the transcription files. Note that a significantly larger error rate was observed for numerals and symbols than for alphabetics. Even with all insertion errors added to the estimate of the alphabetic error, the error rates for numerals and symbols are still significantly higher. One improvement may be to specifically train on common digit strings such as "1989", "80286", and "747" (presently, "1989" is recognized as four separate words instead of the more salient whole). Also, apostrophes are handled incorrectly by expecting only the intra-word stroke version. By expecting both standard stroke orders in words with apostrophes, the system can increase the recognition accuracy of these words significantly. By fixing these simple problems and using BYBLOS's optimizing features, a 10-50% reduction in word error rate may occur.

In this experiment we used a large number of training sentences per writer. Supplying such a large amount of training text may be tiring for just one writer. However, there is some evidence that not as many training sentences per writer are needed for good performance. Furthermore, if good word error rates for the cursive dictation task can be assured, a writer may be willing to spend some time writing sample sentences. A possible compromise is to create a writer independent sytem which can then be adapted to a particular writer with a few sample sentences. With this level of training it may be possible to relax the few restrictions made on the writers in this experiment.

Future experiments will be directed at further reduction of the error rates for the writer dependent task. In addition, writer independent and writer adaptive systems will be attempted. Scalability of the number of training sentences will be addressed along with possible changes to the BYBLOS system to better accomodate handwriting.

6. CONCLUSION

We have shown that a HMM based speech recogition system can peform well on on-line cursive handwriting tasks without needing segmentation of training or test data. On a 25,595 word, 86 symbol, writer dependent task over six writers, an average of 4.2% word error rate and an average of 1.4% character error rate was achieved. With some simple tuning, significant reduction in these error rates is expected. These findings suggest that HMM-based methods combined with statistical grammers will prove to be a very powerful tool in handwriting recognition.

7. ACKNOWLEDGMENTS

The authors wish to thank Long Nguyen and George Zavaliagkos for their help with the BYBLOS system, Tavenner Hall and Brenda Pendleton for their assistance in verifying data, and the Vision & Modeling Group, MIT Media Lab for use of their facilities.

8. REFERENCES

[1] C. Tappert, C. Suen, and T. Wakahara. The State of the Art in On-Line Handwriting Recognition *IEEE T. Pat. Anal. & Mach. Int.*, pp. 787-808, August 1990.

[2] R. Nag, K. H. Wong, F. Fallside, Script Recognition using Hidden Markov Models. In *Proc. ICASSP*, pp. 2071–2074, Tokyo, Japan, 1986.

[3] K. Nathan, J. Bellegarda, D. Nahamoo, E. Bellegarda. On-Line Handwriting Recognition Using Continuous Parameter Hidden Markov Models. In *Proc. ICASSP*, pp. V–121–124, Minneapolis, MN, 1993.

[4] F. Kubala, A. Anastasakos, J. Makhoul, L. Nguyen, R. Schwartz, G. Zavaliagkos. Comparative Experiments on Large Vocabulary Speech Recognition To be presented at *ICASSP*, Adelaide, Australia, 1994.

[5] MADCOW Multi-Site Data Collection for a Spoken Language Corpus. *Proc. DARPA Speech and Natural Language Workshop*, pp. 7–14, Harriman, NY, Morgan Kaufmann Publishers, 1992.

[6] R. M. Schwartz, Y. L. Chow, O. A. Kimball, S. Roucos, M. Krasner, and J. Makhoul Context-Dependent Modeling for Acoustic-Phonetic Recognition of Continuous Speech. *Proc. ICASSP*, pp.1205–1208, Tampa, FL, March 1985.

[7] D. Paul The Design for the Wall Street Journal-based CSR Corpus. *Proc. DARPA Speech and Natural Language Workshop*, pp. 357–360, Morgan Kaufmann Publishers, 1992.