SIGNAL REPRESENTATIONS FOR HIDDEN MARKOV MODEL BASED ON-LINE HANDWRITING RECOGNITION

J.G.A. Dolfing
Philips Research Laboratories
Prof. Holstlaan 4
NL-5656 AA Eindhoven, The Netherlands
Email: dolfing@natlab.research.philips.com

R. Haeb-Umbach
Philips GmbH Forschungslaboratorien
Weißhausstr. 2
D-52066 Aachen, Germany
Email: haeb@pfa.research.philips.com

ABSTRACT

This paper addresses the problem of on-line, writer-independent, unconstrained handwriting recognition. Based on Hidden Markov Models (HMM), which are successfully employed in speech recognition tasks, we focus on representations which address scalability, recognition performance and compactness. 'Delayed' features are introduced which integrate more global, handwriting specific knowledge into the HMM representation. These features lead to larger error-rate reduction than 'delta' features which are known from speech recognition and even require fewer additional components. Scalability is addressed with a size-independent representation. Compactness is achieved with Linear Discriminant Analysis (LDA). The representations are discussed and the results for a mixed-style word recognition task with vocabularies of 200 (up to 99% correct words) and 20,000 words (up to 88.8% correct words) are given.

1. INTRODUCTION

This paper presents the Philips on-line, unconstrained handwriting recognition system. The writer-independent system is based on Hidden Markov Models (HMM) and accepts characters, words and sentences written in discrete, cursive or mixed-style. HMMs are well established in speech recognition and have recently gained attention in handwriting recognition, e.g, [10], [7]. A brief description of our recognition system is given in Section 2. In addition to a description of the baseline system, we concentrate on the problem of representation, i.e., the question of finding a suitable representation of a portion of the scribble sequence recorded on a tablet specifically to enhance writing size invariance, recognition performance and compactness.

A feature vector should contain all relevant information for the subsequent recognition, be insensitive to irrelevant variations, and at the same time have a low vector dimension. One of the irrelevant variations is the writing size. In case of word-based input where the whole word is captured before processing, writing size is often explicitly normalized as in [7] and [11]. A more flexible approach which allows simultaneous writing and recognition is discussed in Section 3 where we discuss a size-dependent representation (frames) and a size-independent representation (segments). The inclusion of approximations in the time difference of the feature sequence, so-called delta features, are known from speech recognition to improve recognition performance [8]. In Section 4 we introduce the concept of *delayed* features and compare them with delta features. Delayed features provide a flexible framework to integrate handwriting specific knowledge into the representation by expressing structural relations between the current and previous observation vectors. Finally, LDA has been employed in the training and recognition process as described in Section 5, similar as in [4] for speech recognition, to improve performance and efficiency.

2. BASELINE SYSTEM

The platform for capturing handwriting is a Philips proprietary tablet called Philips Advanced Interactive Display (PAID) consisting of an LCD plus orthogonal sensors for pen and finger input sampling (x,y,p) with up to 200 pps. This tablet is connected to a PC with pen-enhanced Unix or PenWindows. Data is filtered and, depending on representation, spatially resampled.

Each character is modeled by a left-to-right hidden Markov model with loop, forward and skip transitions between the states. Optionally, the models can be extended with a pause state after the word to catch delayed strokes which are otherwise not processed. The observation probabilities are continuous mixtures of Gaussian densities with density-specific diagonal cov-

0-8186-7919-0/97 $10.00

ariance matrices. Up to 32 densities per mixture are allowed. Training of the HMM parameters is done by using the Maximum Likelihood criterion and applying the Viterbi approximation [8].

Recognition is based on the one-stage beam search algorithm using a tree-organized dictionary [5]. All knowledge sources, i.e., the pen signal, the dictionary, and, for some experiments, a language model, are applied at once, thus avoiding premature decisions. Hypotheses pruning is applied for efficiency.

For the experiments reported here the training data consisted of more than 10000 handwritten words from about 60 writers of several nationalities from on-site collected data and Unipen training data [3].

2.1. Performance

Essentially, the same recognizer was employed for the recognition of characters, words or whole sentences for a 200 words and 20,000 words dictionary. While the 20,000 words dictionary is a random subset of the 60,000 most common English words, the 200 words vocabulary is a subset where all character classes occur approximately equally often. Average word length of vocabularies is 6.5 and 8 characters for the 200 and 20,000 words vocabulary, respectively. The HMM framework is able to simultaneously determine the optimal segmentation and carry out the decoding. We obtained word recognition rates up to 99% for the 200 words vocabulary and up to 90% correct words for a 20,000 words vocabulary without language model. These results compare well with other published results for unconstrained handwriting recognition [7], [11], [6]. Initial experiments on sentence recognition like in [10] have also been conducted.

3. WRITING SIZE INVARIANCE

Like writing speed and sampling rate, writing size is often explicitly normalized like in [7] and [11]. This approach is very suitable for isolated word recognition tasks where all input is available before normalization but less suitable for sentence recognition where writing and recognition are done simultaneously. As an alternative we investigated size-independent representations.

Two alternatives were compared for grouping samples into blocks of which a feature vector is computed. In the case of *segments*, the block boundaries are defined by the condition that the vertical handwriting speed is zero: $v_y = 0$. The location of these points within a character is invariant with respect to the handwriting size. In combination with writing size independent features, this results in a writing size independent representation. In contrast, a *frame* consists of a fixed number of consecutive, resampled points. Resampling the pen trajectory is necessary to obtain equispaced points and thus compensate for writing speed variations.

In both cases the sample blocks used to compute adjacent feature vectors were chosen to overlap by 50%. For average writing size (see below) there were on the average about ten frames (seven segments) per character corresponding to seven (five) states per model.

The feature vectors were identical for frames and segments and contained 13 low-level, size-independent features like aspect ratio, curvature, five angles, a pen-down feature. Additionally, four delayed features (see next section) were used with different delays for frames and segments.

To test writing size dependence, we asked 10 writers to write a set of 50 words in four different sizes (scale: 0.5, 1, 2, 4) where scale 1 corresponds to the average writing size of the training data. Writers were instructed to write lowercase but unconstrained otherwise. The resulting set of 4 times 500 words was represented by either frames or segments and recognized using a 200 and 20K word vocabulary. The results presented in Table 1 clearly show that, while frames have a better peak performance (of up to 99% words correct), the segments are essentially independent of the writing size.

Table 1: Comparison of frames and segments for four different writing sizes. The table contains recognition rates in % words correct for a 200 and 20,000 word dictionary.

	200 word dictionary			
Size	0.5	1	2	4
Frame	82.1	99.0	68.8	2.7
Segment	96.8	96.8	98.0	97.1
	20,000 word dictionary			
Size	0.5	1	2	4
Frame	55.5	90.2	33.7	0.0
Segment	82.5	83.3	85.3	81.8

4. DELAYED AND DELTA FEATURES

It is known that feature events that describe consistent trends in the handwriting over several points can improve recognition accuracy. One method is to splice adjacent frames to an enlarged feature vector [1]. Here we investigated alternative approaches.

Let o_t denote a feature of the current frame t. A way to describe the dynamics of the signal, which has been adopted from speech recognition [8], is the use of *delta* features, i.e., approximations to the derivatives of the observation vector with respect to time, e.g.,

$\Delta o_t = \frac{1}{2}(o_{t+1} - o_{t-1})$. Combination of o_t and Δo_t yields a new enlarged feature vector o_t'.

We developed the novel concept of, what we called, *delayed* features, which are used to measure the spatial dynamics of the signal. Sample realizations of structural handwriting knowledge based on delayed features are *positional*, *size* and *overlap* relations. Knowledge is modeled by relating the current segment (or frame) with previous or future segments (frames).

An example of positional relation is the angle between the line connecting the center-of-gravities (cog) of subsequent feature vectors and the x-axis. For example, a delay $n = 2$ yields the features $\sin(\text{angle}(cog(o_t), cog(o_{t-2})))$, $\cos(\text{angle}(cog(o_t), cog(o_{t-2})))$ which describes the change of writing direction between feature vectors o_{t-2} and o_t. In the same way, we can introduce a new feature which describes size relation, e.g., delay $n = 2$ yields $length(o_t)/(length(o_t) + length(o_{t-2}))$ where $length(o_t)$ denotes the path-length of the segment or frame at time t. Furthermore, the concept of delayed features allows to compute overlap relations similar to the "hat-feature" in [9].

Based on the 500 normal-sized words of the scalability test, we compared the performance of a baseline representation (13 components) to feature vectors which included delta (baseline + 13 deltas) and delayed features (baseline + 6 delayed features), respectively. The six delayed features are constructed from three angles with delay 1,2 and 4 representing positional relation. Table 2 shows that the augmented feature vector clearly outperforms the baseline representation, delayed features performing better than delta features. In the case of the delayed features, fewer additional vector components are required to attain the performance improvement.

Table 2: Recognition rates (in % words correct) for segment type of feature vectors for a 200 word and a 20,000 word dictionary.

#features	13 Baseline	13+6 delayed =19	13+13 delta =26
200 W	92.5	97.6	97.3
20,000 W	72.0	86.7	81.9

5. LINEAR DISCRIMINANT ANALYSIS

Linear discriminant analysis (LDA) [2] is a well known technique in statistical pattern classification for compressing the information contents (with respect to classification) of a feature vector by a linear transformation.

The resulting feature vector is decorrelated, ordered, and maximally compact. The first property is advantageous, since in our HMMs we employ diagonal covariance matrices, see Section 2. The second property means that the features are ordered according to decreasing eigenvalue. The first features, i.e., those with the largest eigenvalues, contribute most for class separability. Finally, the third property states that for any subset of the features 1 to k the sum of the eigenvalues is maximum. No other subset of the same size k of features or linear combinations thereof has a larger sum.

LDA has been employed in the training and recognition process similar as described in [4] for speech recognition. Training is carried out in three steps:

- First an ordinary training is carried out. This yields a segmentation, i.e., a class label for each feature vector. Note that we defined the classes in the LDA sense to be HMM states.
- Next the within and between class scatter matrices are computed and from them the LDA transformation is obtained by solving an eigenvalue problem [2].
- Finally, a completely new training is conducted on the LDA-transformed feature vectors. Optionally, the dimension of the transformed feature vector can be reduced by discarding the least important rows of the LDA transformation matrix.

Two aspects of the LDA transform are investigated. First, the performance improvement due to the LDA transform is tested while the feature vector size before and after transformation remains the same. Second, the error rate as a function of the feature vector size after transformation is investigated.

First, Table 3 summarizes recognition results with LDA in an otherwise unchanged experiment. Comparing this table with the results without LDA in Table 2 shows a clear improvement.

Table 3: Recognition rates (in % words correct) for segment type of LDA transformed feature vector for a 200 word and a 20,000 word dictionary. Full dimension of LDA transformed feature vector.

#features before LDA	13 Baseline	13+6 delayed =19	13+13 delta =26
200 W	96.3	97.8	97.4
20,000 W	81.0	87.3	86.3

Second, Figure 1 shows the relation between the number of features retained after transformation and

performance. For both delta and delayed features, a performance peak is reached after dropping about four features. A peak performance of 88.8% words correct is reached.

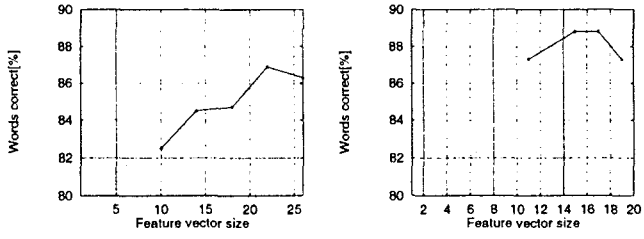

Figure 1: [%] words correct versus feature vector size after the LDA for delta (left figure) and delayed (right figure) representation prior to LDA (20,000 words vocabulary)

6. CONCLUSION

We have described a number of different representations in order to address the goals of writing size independence, integration of handwriting knowledge based on delayed features and a robust, compact representation. We have shown that the writing size independence implicit in the segment based representation offers an attractive alternative to the usual word-based size normalization at the cost of a small peak performance decreasement. The tested representation alternatives (delta, delay and LDA) each contribute a performance improvement. In the tested configuration for a segment based recognizer with 20K vocabulary, the combined improvements yield a performance improvement from 72% to 88.8% which means that the error-rate is more than halved due to representation enhancements only.

It is expected that even larger improvements are possible since the full potential of delayed features has not been exploited yet. Further, the combination of delta and delayed features may also be worthwhile studying since the 'extra' information in the representation is different for deltas and delays.

7. REFERENCES

[1] E.J. Bellegarda, J.R. Bellegarda, D. Nahamoo, and K.S. Nathan. A fast statistical mixture algorithm for on-line handwriting recognition. *IEEE Transactions on pattern analysis and machine intelligence*, 16(12):1227–1233, December 1994.

[2] K. Fukunaga. *Introduction to Statistical Pattern Recognition, Second Edition.* Academic Press, New York, 1990.

[3] I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, and S. Janet. Unipen project of on-line data exchange and recognizer becnhmarks. In *12thInternational Conference on Pattern Recognition*, volume 2, pages 29–33, 1994.

[4] R. Haeb-Umbach and H. Ney. Linear discriminant analysis for improved large vocabulary continuous speech recognition. In *International Conference of Acoustics, Speech and Signal processing*, pages 13–16, March 1992.

[5] R. Haeb-Umbach and H. Ney. Improvements in beam search for 10,000-word continuous-speech recognition. *IEEE Transactions on Speech and Audio Processing*, 2(2):353–356, April 1994.

[6] S. Manke, M. Finke, and A. Waibel. Npen++: A writer independent, large vocabulary on-line cursive recognition system. In *International Conference on Document Analysis and Recognition*, volume 1, pages 403–408, 1995.

[7] K.S. Nathan, H.S.M. Beigi, J. Subrahmonia, G.J. Clary, and H. Maruyama. Real-time on-line unconstrained handwriting recognition using statistical methods. In *International Conference of Acoustics, Speech and Signal processing*, pages 2619–2622, 1995.

[8] L. Rabiner and B.H. Juang. *Fundamentals of speech recognition.* Prentice hall, first edition, 1993. ISBN 0-13-015157-2.

[9] M. Schenkel, I. Guyon, and D. Henderson. On-line cursive script recognition using time-delay neural networks and hidden markov models. In *International Conference of Acoustics, Speech and Signal processing*, volume 2, pages 637–640, 1994.

[10] T. Starner, J. Makhoul, R. Schwartz, and G. Chou. On-line cursive handwriting recognition using speech recognition methods. In *International Conference of Acoustics, Speech and Signal processing*, volume 5, pages 125–128, 1994.

[11] H. Weissman, M. Schenkel, I. Guyon, C. Nohl, and D. Henderson. Recognition-based segmentation of on-line run-on handprinted words: input vs output segmentation. *Pattern recognition*, 27(3):405–420, 1994.