A DISCRETE PARAMETER HMM APPROACH TO ON–LINE HANDWRITING RECOGNITION

*Eveline J. Bellegarda, Jerome R. Bellegarda,***
David Nahamoo, and *Krishna S. Nathan*

IBM Research
T.J. Watson Research Center
Yorktown Heights, New York 10598

ABSTRACT

One area where on-line handwriting recognition technology is most critical is the domain of small portable platforms. Because such platforms have limited resources, it is not presently practical to consider a continuous parameterization for the hidden Markov models used in the recognition. On the other hand, discrete parameter techniques such as used in speech recognition are difficult to apply, because there is no well-understood handwriting equivalent to phonological rules. A possible solution is to extract this information directly from the data, by constructing an alphabet of sub-character, elementary handwriting units. The performance of this method is illustrated on a discrete handwriting recognition task with an alphabet of 81 characters.

I. INTRODUCTION

An automatic handwriting recognition system can be viewed as aiming to recover the character sequence most likely to correspond to some given handwritten evidence. This approach assumes the existence of probabilistic models for estimating the likelihoods of candidate character sequences. Accordingly, in the past few years there has been a growing interest in probabilistic techniques for (both off-line and on-line) handwriting recognition; see, e.g., [1] – [5].

At ICASSP'93, we disclosed an algorithm for on-line handwriting recognition based on tied mixture continuous parameter hidden Markov modeling [3]. This approach relies on a decomposition of the total character sequence likelihood into a language model contribution and a handwriting channel model contribution. To characterize the handwriting channel, we derive a left-to-right hidden Markov model (HMM) for each allograph sufficiently represented in the training data. These allographic models are then concatenated appropriately to represent whole character sequences. This approach proved useful to tackle a large alphabet handwriting recognition task.

At ICASSP'94, we enhanced this algorithm to ensure that the HMMs closely track the allograph trajectories in the underlying feature space (referred to in [3] as the *chirographic space*). This capability is necessary to adequately account for the numerous (and sometimes severe) allograph deformations inherent to unconstrained handwriting recognition. To correctly capture realizations in the chirographic space, a greater number of continuous parameter prototype distributions, or prototypes for short, is usually required. This, however, may not be a realistic option due to the limited size of the available training data. An alternative solution is to improve the partition of the underlying feature space based on the concept of contextual supervision [5]. This leads to enhanced allograph representations by relating the HMM allographic models to their manifestations in chirographic space. We showed in [5] that this approach has two main advantages: (i) it better accounts for both intra-speaker and inter-speaker variations in handwriting; (ii) it makes better use of the available training data than its unsupervised counterpart.

This paper explores the consequences of the supervision process introduced in [5] to the problem of HMM parameterization. One area where on-line handwriting recognition technology is most critical to product success is the domain of small portable platforms, where keyboard entry is not a viable input option. Because both CPU and memory resources are severely constrained on these platforms, however, it is difficult to accommodate continuous parameter approaches like in [3] – [5]. Discrete parameter techniques such as pioneered in speech recognition over the past ten years would better fit the requirements. Unfortunately, they are difficult to apply, because there is no well-understood handwriting equivalent to phonological rules. A possible solution is to extract this information directly from the data. More specifically, the supervised framework described in [5] is used to construct an alphabet of sub-character, elementary handwriting units on which to base the discrete parameterization.

The paper is organized as follows. In the next section we discuss the issue of HMM parameterization, and in Section III we briefly review the concept of su-

*J.R. Bellegarda is now with the Advanced Technology Group, Apple Computer Inc., Cupertino, California 95014.

0-7803-2431-5/95 $4.00 © 1995 IEEE

pervision as introduced in [5]. Section IV describes in greater details how to construct a data-driven alphabet of elementary handwriting units. Finally, in Section V we present experimental results obtained on a 81-character alphabet, discrete handwriting task.

II. PARAMETERIZATION

From a practical point of view, the fundamental limitation of the recognition system described in [5] lies in the choice of HMM parameterization. By using a continuous parameter framework, we force structural assumptions on the output distributions, which can only be alleviated by increasing the number of parameters in the system. This, in turn, tends to require a greater amount of training data, not to mention more computational power. It would be much more expedient to select a discrete parameterization for the chirographic HMMs. Then the output probabilities would not be constrained to follow a particular distribution (such as Gaussian), thus reducing the overall number of parameters and thereby the requirements on data and CPU.

The problem, however, is the loss of information due to vector quantization, which is guaranteed to occur in a discrete parameterization. To minimize this loss, the vector quantizer must be designed very carefully. In particular, this raises the issue of finding a suitable alphabet of handwriting units to perform the discrete parameterization. Using whole characters as units, for example, is not likely to produce a good vector quantizer due to inherent variations in handwriting, writer variations, as well as systematic variations in the features due to changes in the handwriting context. This is precisely this realization which prompted us to select a continuous parameter approach in the first place [3].

A solution to the above dilemma is to further exploit the supervision process introduced in [5]. Recall from [5] that in the course of constructing supervised chirographic prototypes, we build a clustering tree which completely exposes the relevant interrelationship between all potential clusters. This clustering tree is then pruned, possibly using a different criterion, to determine the final clusters, i.e., the desired chirographic prototypes. This pruning is normally adjusted to achieve a suitably fine partition of the underlying feature space. Clearly, it is also possible to prune the tree further, so as to obtain a coarse partition of the space. The resulting distribution may not be adequate to define a rich enough set of prototypes. On the other hand, it represents a way to characterize broad regions in a supervised, data-driven fashion.

We interpret each such region as the manifestation of an elementary (sub-character) unit of handwriting, much more specific than the character itself. This has immediate implications for discrete parameter modeling: it suffices to use the collection of such (data-driven) elementary units as the handwriting alphabet used in the parameterization. Note that each of these units can be easily expressed, through the above clustering tree, as a mixture of prototype distributions. This approach is therefore analogous to a strategy commonly used in speech recognition, with the above elementary handwriting unit playing the role of a phoneme. To pursue the analogy further, the clustering tree then embodies the equivalent of the phonological rules used in speech recognition.

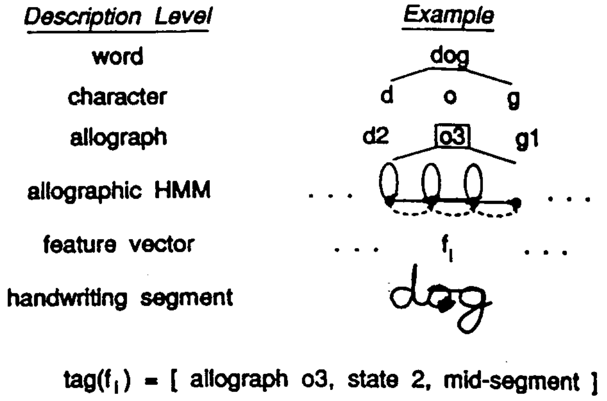

Fig. 1. Illustration of Tagging Procedure.

III. SUPERVISION

As emphasized in [5], the main purpose of supervision is to suitably partition the underlying chirographic space, so that a proper representation is obtained for the chirographic realization of each allograph. Bottom-up clustering leads to a hierarchy of allographic chirographic realizations, and pruning finalizes each of the elementary units to achieve a suitable level of generality and robustness. For the sake of flexibility, the two steps need not operate under the same criterion.

To allow for supervision, some preliminaries must be satisfied. As an initial step, we therefore assume that some handwriting has been recorded, signal processed, and Viterbi aligned against suitable allographic Markov models (such as those derived in [3] and described in more details in [4]). On the basis of the Viterbi alignment, each feature vector is tagged with an index which unambiguously identifies the following: (a) the identity of the associated allograph A; (b) the location L_S of the state of the allographic model A against which the feature vector is aligned; and (c) the location L_F of the feature vector within the handwriting segment corresponding to the allographic model A where the feature vector is aligned.

To fix ideas, the above tagging mechanism is illustrated in Fig. 1, which examines the different descriptions available for the word "dog," in order of increasing level of detail. The word can be expanded at the character level, or, taking the context into account, at the allograph level. Each allograph can in turn be associated with an allographic HMM, cf. [3] and [4]. Further down, we arrive at the feature vector level. In the example of Fig. 1, the feature vector f_i is part of the allograph $o3$ and has been Viterbi aligned against the second state of the associated allographic HMM. In addition, f_i also embodies a portion of handwriting which has occurred in approximately the middle (shaded area) of the handwriting segment representing "o." Thus, in this example, we could write $A = o3$, $L_S = 2$, and $L_F = 1/2$.

Clearly, two feature vectors tagged with the same index through the above tagging procedure share extremely similar chirographic properties. In other words, each tag represents a very specific chirographic sub-event, which in turn can be used to supervise the prototype and elementary unit construction. Note that the main purpose of the measure L_F is to identify transient effects at the boundaries of each handwriting segment. Because this measure is essentially uninformative within each character, a practical implementation of (c) may threshold the information L_F away from the boundaries. In the experimental results reported in this paper, we used a threshold of 4 frames. Thus, all positions L_F falling more than 4 frames into the handwriting segment and more than 4 frames before the end of the handwriting segment were assigned an identical value (1/2). We found this useful to keep the number of tags manageable.

IV. ELEMENTARY UNITS

After all training vectors have been appropriately tagged, the complete algorithm proceeds as follows: (i) for each allographic model, pool together all the frames that have been aligned to each individual state, and compute their centroid and their count; (ii) construct a bottom-up binary clustering tree which completely exposes the relevant inter-relationships between all potential clusters; (iii) prune this clustering tree according to some appropriate criterion, so as to retain a predetermined number of clustering leaves; (iv) define an elementary handwriting unit for each of the clustering leaves found in (iii).

Note that all the feature vectors that have been aligned with instances of each 3-tuple $T = (A, L_S, L_F)$ belong to the same specific chirographic sub-event in the feature space. As in [5], we refer to the centroid of these vectors as an *anchor point* of the chirographic sub-event. By definition, there are as many anchor points as there are 3-tuples (A, L_S, L_F). To these anchor points, we apply an iterative clustering procedure whose output is a binary tree such as illustrated in Fig. 2. The anchor points are at the bottom of the tree, two of them being shown as T_1 and T_2. Each node of the tree (like P) exposes an elementary "closeness" relationship between its two descendants.

As in [5], the goal is to establish a hierarchy of potential clusters. At the root of the tree, there is only one cluster containing all the feature vectors. At the bottom of the tree, there are as many clusters as there are anchor points, each cluster representing a very specific chirographic sub-event. Alternatively, one can take various cuts across the middle of the tree, two of which are shown in Fig. 2 as the heavy solid and dashed curves. These cuts correspond to different cluster sets with various degrees of detail. For example, the clusters associated with the dashed curve will be intrinsically smoother than those associated with the solid curve, thus inherently less descriptive but more robust. For the purpose of deriving chirographic prototypes it may have been sufficient to iterate pruning until the solid curve of Fig. 2 was reached (cf. [5]). Here, however, it may be more suitable to iterate until the dashed curve of Fig. 2 is reached.

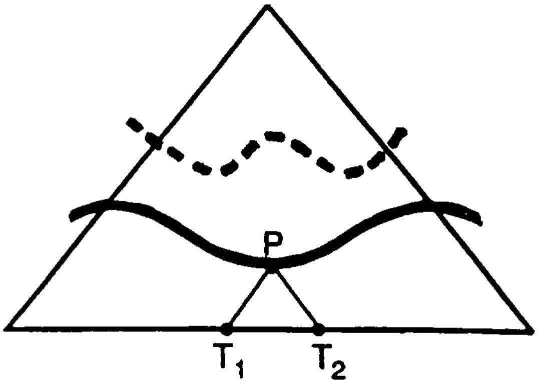

Fig 2. General Clustering Tree Structure With Two Possible Cluster Sets.

Once this is done, the statistics (mean vectors and covariance matrices) obtained for all the anchor points are propagated upwards to the leaves of the pruned clustering tree just constructed. This defines a (Gaussian) distribution at each leaf of the tree, which embodies one elementary handwriting unit. Alternatively, it defines a mixture distribution, incorporating the appropriate prior probabilities based on cluster sizes, which covers the entire inventory of allographs. This distribution uniquely defines the set of elementary handwriting units sought.

V. EXPERIMENTAL RESULTS

We considered a handwriting recognition task with a 81-character alphabet comprising upper and lower case letters, digits, and 19 special symbols (mathematical, punctuation). Such a large alphabet makes the task quite challenging due to numerous shape confusions (e.g., "quote" vs. "comma"). The data were collected on a transparent electronic tablet which had a resolution of 254 points/in. On the average, 60 coordinate pairs per character were captured by the tablet. Two series of experiments were conducted, addressing writer-dependent (WD) and writer-independent (WI) recognition, respectively. For the WD experiments, eight writers each provided 300 words of training data

Writer	Continuous Parameter Modeling	Discrete Parameter Modeling
MAR	4.9 %	4.7 %
HIL	18.1 %	18.3 %
VIV	10.3 %	10.2 %
JOA	9.9 %	9.8 %
ACY	21.7 %	22.0 %
NAN	9.1 %	9.1 %
LOY	7.4 %	7.3 %
SAB	6.4 %	6.2 %
Average	10.9 %	10.9 %
CPU Index	100	5

Table I. Writer–Dependent Character Error Rates With Continuous and Discrete Parameterization.

and 150 words of test data. For the WI experiments, the training data for these eight writers was pooled together, and four additional writers (unrelated to the previous ones) each provided 150 words of test data. Since our primary goal was to evaluate the goodness of the algorithms rather than get absolute recognition numbers, no language model was used for either experiment. Note that, as a result, inherently ambiguous characters such as "zero" and "oh" cannot be distinguished.

The error rates obtained using the discrete parameter approach proposed in this paper were compared to the error rates obtained using the (supervised) continuous parameter method of [5], with the same training data. Also compared were the amounts of CPU time necessary to run the experiments in each case, normalized to that obtained on the baseline (index 100). In the WD experiments, eight distinct clustering trees (one for each writer) were grown and pruned as described earlier. In the WI experiment, only one clustering tree was derived from the pooled data. The results are summarized in Tables I and II for WD and WI recognition, respectively.

Over all the speakers we consistently observe a large decrease in the CPU time required to run the experiment, while the average error rate remains essentially unchanged. This reduction varies from a factor of 14 (WI) to a factor of 20 (WD). These results show that discrete parameter HMMs are promising for on-line handwriting recognition on small platforms with limited resources.

VI. CONCLUSION

We have described a discrete parameter hidden Markov modeling approach suitable for handwriting recognition systems using allographic models. The discrete labels used for vector quantization come from an alphabet of sub-character, elementary handwriting units. This alphabet is derived directly from the underlying chirographic space by incorporating supervision to relate the HMM allographic models to their chirographic manifestations.

The overall procedure is decoupled into a clustering phase followed by a pruning phase. This way, all the general inter-relationships between various chirographic sub-events are uncovered once, while it is possible to customize both the alphabet of elementary units and the underlying chirographic prototypes according to the available training data. This makes for an efficient, streamlined recognition system, as evidenced by a significant decrease in CPU requirements with respect to a baseline system using a continuous parameterization.

Writer	Continuous Parameter Modeling	Discrete Parameter Modeling
MAL	20.3 %	20.4 %
BLA	16.4 %	16.4 %
SAM	26.0 %	26.5 %
WAR	12.7 %	13.2 %
Average	18.9 %	19.1 %
CPU Index	100	7

Table II. Writer–Independent Character Error Rates With Continuous and Discrete Parameterization.

REFERENCES

[1] E. Levin and R. Pieraccini, "Dynamic Planar Warping for Optical character Recognition," in *Proc.* 1992 *ICASSP*, Vol. 3, San Francisco, CA, pp. 149–152, April 1992.

[2] Y. He, M.Y. Chen, and A. Kundu, "Handwritten Word Recognition using HMM with Adaptive Length Viterbi Algorithm," *Proc.* 1992 *ICASSP*, Vol. 3, San Francisco, pp. 153–156, April 1992.

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

[4] E.J. Bellegarda, J.R. Bellegarda, D. Nahamoo, and K.S. Nathan, "A Probabilistic Framework for the Recognition of On–Line Handwriting," in *Proc. 3rd Int. Workshop on Frontiers in Handwr. Reco.*, Buffalo, NY, pp. 225–234, May 1993.

[5] J.R. Bellegarda, D. Nahamoo, K.S. Nathan, and E.J. Bellegarda, "Supervised Hidden Markov Modeling for On–Line Handwriting Recognition," in *Proc.* 1994 *ICASSP*, Vol. V, Adelaide, Australia, April 1994.