Using HMM Based Recognizers for Writer Identification and Verification

Andreas Schlapbach and Horst Bunke
Department of Computer Science, University of Bern
Neubrückstrasse 10, CH-3012 Bern, Switzerland
{schlpbch, bunke}@iam.unibe.ch

Abstract

In this paper, we use HMM based recognizers for the identification and verification of persons based on their handwriting. For each writer, we build an individual recognizer and train it on text lines of that writer. This gives us recognizers that are experts on the handwriting of exactly one writer. In the identification or verification phase, a text line of unknown origin is presented to each of these recognizers and each one returns a transcription that includes the log-likelihood score for the considered input. These scores are sorted and the resulting ranking is used for both identification and verification. In an identification experiment in 96.56% of all cases the writer out of a set of 100 writers is correctly identified. Second, in a verification experiment using over 8,600 text lines from 120 writers an Equal Error Rate (EER) of about 2.5% is achieved.

Keywords: writer identification, writer verification, off-line handwriting, HMM based handwriting recognition.

1. Introduction

Writer identification is the task of determining the author of a sample handwriting from a set of writers [15]. Related to this task is writer verification, i.e., the task of determining whether or not a handwritten text has been written by a certain person. If any text may be used to establish the identity of the writer the identification task is text independent. Otherwise, if a writer has to write a particular predefined text to identify himself or herself the identification task is text dependent. Writer identification can be performed on-line, where temporal and spatial information about the writing is available, or off-line, where only a scanned image of the handwriting is available. This paper addresses the problem of text independent writer identification and verification using off-line data. Given a text line (some examples are given in Fig. 1) the system must either identify the author of the text line or verify whether a text line is from a particular writer.

Much progress has been made in handwriting recognition in the last decades [16]. In recent years Hidden Markov Models (HMMs) have become the predominant approach for isolated word and general text recognition [6]. In this paper, we use HMM based handwriting recognition systems for the purpose of writer identification and verification. For each writer in the considered population, an individual HMM based handwriting recognition system is trained using only data from that writer. Thus for N different writers we obtain N different HMMs. They all have the same architecture, but their parameters, i.e., transition and output probabilities, are different as the systems have been trained on different data. Intuitively, each HMM can be understood as an expert specialized in recognizing the handwriting of one particular person. In the recognition phase, a text line of unknown identity is presented to each HMM based recognizer. Each recognizer outputs a transcription of the input together with a recognition score in terms of a log-likelihood value. These outputs are sorted in decreasing order of the recognition scores, giving us a ranking of all systems. Based on this ranking, we can address the task of identifying the writer of a text line or of verifying whether a text line has actually been written by the person who claims to be the writer. We assume that correctly recognized words have a higher score than incorrectly recognized words, and that the recognition rate of a system is higher on input from the writer the system was trained on than on input from other writers.

For the first task of writer identification, we use the ranking to decide who has written the input text line in the following way: we opt for the writer whose recognizer produces the highest score. The second task addressed in this paper is the verification of handwritten text lines. A verification system must decide whether a text line with a claimed identity was in fact written by this person or not. In the second case the person is called an impostor [4]. Impostor attempts can be divided into unskilled forgeries, where the impostor makes no effort to simulate a genuine handwriting, and skilled forgeries, where the impostor tries to imitate the handwriting of a client as closely as possible [15]. In this paper, we address the former problem. To

Proceedings of the 9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004)
0-7695-2187-8/04 $20.00 © 2004 IEEE

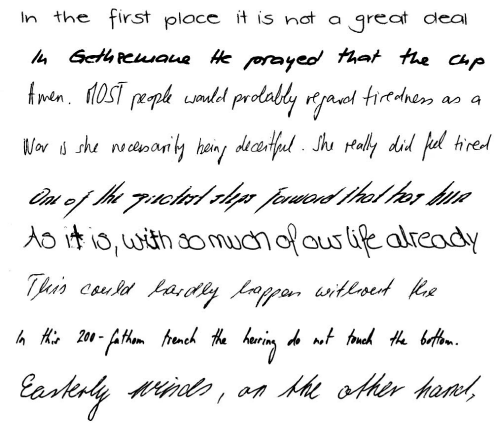

Figure 1. Text Line Examples.

simulate impostor attempts, the handwriting of a person unknown to the system is taken and it is claimed that it is from one of the N writers the system was trained with. A set of confidence measures is used to formulate a verification criterion. The confidence measures are calculated based on the difference of the log-likelihoods of the N-best ranked writers, possibly including the claimed writer, or the N-best ranked competing writers.

This paper is structured as follows. The next section presents related work. Our handwritten text line recognizers are described in Section 3. In Section 4 we show how we combine them to build a writer identification and verification system for hand written text lines using a set of confidence measures. The underlying database and the results of our experiments are presented in Section 5. Finally, in Section 6 we conclude the paper.

2. Related Work

Surveys covering work in automatic writer identification and signature verification until 1993 are given in [9, 15]. Writer identification can be understood as a classification problem where a word, text fragment, or text is to be assigned to one out of a number of possible writers. Recently, different approaches to writer identification have been proposed. Said et al. [19] treat the writer identification task as a texture analysis problem. They use global statistical features extracted from the entire image of a text using multi-channel Gabor filtering and grey-scale co-occurrence matrix techniques.

Cha et al. [7] address the problem of writer verification, i.e., the problem of determining whether two documents are written by the same person or not. In order to identify the writer of a given document, they model the problem as a classification problem with two classes, *authorship* and *non-authorship*. Given two handwriting samples, one of known and the other of unknown identity, the distance between two documents is computed. Then the distance value is used to classify the data as positive or negative.

Zois et al. [22] base their approach on single words by morphologically processing horizontal projection profiles. The projections are derived and processed in segments in order to increase the discrimination efficiency of the feature vectors which are then classified using either a Bayesian classifier or a neural network.

In Hertel et al. [8] a system for writer identification is described. The system first segments a given text into individual text lines and then extracts a set of features from each text line. The features are subsequently used in a k-nearest-neighbor classifier that compares the feature vector extracted from a given input text to a number of prototype vectors coming from writers with known identity. In a 50 writers experiment, in 96.4% of all cases the writer is correctly identified.

Bulacu et al. [5] use edge-based directional probability distributions as features for the writer identification task. They introduce edge-hinge distribution as a new feature. The key idea behind this feature is to consider two edge fragments in the neighborhood of a pixel and compute the joint probability distribution of the orientations of the two fragments. This feature performs better than other features they evaluated.

In a set of papers [2, 3, 14] graphemes are proposed as features for describing the individual properties of handwriting. Furthermore, it is shown that each handwriting can be characterized by a set of invariant features called the writer's invariants. These invariants are detected using an automatic grapheme clustering procedure.

Leedham et al. [10] present a set of eleven features which can be extracted easily and used for the identification and verification of documents containing handwritten digits. These features are represented as vectors and by using the Hamming distance measure and determining a threshold value for the intra-author variation a high degree of accuracy in authorship detection is achieved.

In [20], we have presented an off-line handwriting identification system using HMM based recognizers. We tested our system using over 2,200 text lines coming from 50 writers and have in 94.47% of all cases correctly identified the writer. Using a simple confidence measure, we achieved an error rate of 0% by rejecting 15% of the results. This paper presents a number of extensions to this work. First, the number of writers is extended from 50 to 100 in the first, and from 50 to 120 in the second set of experiments (see Section 5.2 and 5.3) coming from a subset of the IAM database [12] different than the one used in [20]. Second, rather than considering just a single confidence measure, a number of such measures and related rejection strategies are investigated.

Third, while [20] was restricted to writer identification, we also address the problem of writer verification in this paper.

3. HMM Based Recognizers for Handwritten Text Lines

The system we present in this paper uses HMM based recognizers that are designed and optimized for the task of handwritten text line recognition. These recognizers are derived from the system described in [11].

In a number of preprocessing steps, the text lines presented to the recognizers are normalized. The following normalization operations are applied: The slant and the skew angle of the text lines are corrected, base line localization is performed and the text lines are normalized with respect to the width of the text line. A sliding window which moves from left to right over the text lines, extracts nine features, three global and six local ones. The global features are the fraction of black pixels in the window, the center of gravity and the second order moment. The local features represent the position of the upper and the lower-most pixel, the number of black-to-white transitions in the window, and the fraction of black pixels between the upper and lower-most black pixel. Using these features, an input text line is converted into a sequence of 9-dimensional feature vectors. A more detailed description of the feature extraction process is given in [11].

For each upper and lower case character an individual HMM is built. Additionally, we model frequent punctuation marks, such as full stop, colon and space. Other, infrequent punctuation marks are mapped to a special garbage model. Each character HMM consists of 14 states connected in a linear topology. These character models are concatenated to word models which in turn are concatenated to model a complete text line.

We train the system by applying the Baum-Welch algorithm [17]. The following training strategy is applied. In the first step, a single Gaussian output distribution for each state is used. Each model is trained with four iterations. Then in the second step, the number of Gaussian mixture components is increased. This is implemented by splitting the Gaussian distribution with the highest weight. The mean vectors of the two new Gaussian distributions are the mean of the original Gaussians ± 0.2 times the standard deviation of the original distribution [21]. Then in the third step, we again train each model in four iterations using the new mixture components. Steps 2 and 3 are repeated until the desired number of Gaussian mixture components is reached. Preliminary experiments have shown that using four Gaussians mixture components leads to good recognition results.

For recognition, the Viterbi algorithm is used. Presented with a text line, a recognizer produces a sequence of words together with their log-likelihood scores. Summing up the scores of all words gives us the log-likelihood score of a text line.

4. A Writer Identification and Verification System Using Text Line Recognizers

4.1. Writer Identification

For each writer in the considered population of clients, a text line recognizer as described in the previous section is built and trained with data coming from that writer only. As a result of the training procedure, we get a recognizer for each writer that is an expert on the handwriting style of that particular person.

For the task of writer identification, we present a text line of an unknown writer to each of the trained recognizers. Each recognizer outputs a transcription of the input text line together with its log-likelihood score. The log-likelihood scores are sorted in descending order and the input text line is assigned to the writer with the highest ranked score. Using a confidence measure [13] enables us to implement a rejection mechanism: if the confidence measure of a text line is above a given threshold, the system returns the identity of the text line with the highest ranked score; otherwise the system rejects the input. In a previous paper [20], we have used the difference of the log-likelihood of the best and second best ranked writer, normalized by the length of the text line, as a confidence measure for each text line. In this paper we extend this idea, inspired by the cohort score normalization technique used in the field of speaker verification [1, 18]. Instead of only using the log-likelihoods of the first two ranks we can use the log-likelihood scores of the first N ranks to calculate a confidence measure. We define the confidence measure, $cm_{text\ line}$, for a text line as follows:

$$cm_{text\ line} = \frac{l_1 - l_{avg}}{text\ line\ length} \tag{1}$$

where

$$l_{avg} = \frac{1}{N} \sum_{j=1}^{N} l_j \tag{2}$$

or

$$l_{avg} = \frac{1}{N} \sum_{j=2}^{N+1} l_j \tag{3}$$

By using alternatively Eq. 2 or Eq. 3, we can either include the first ranked system in the sum of log-likelihoods or not. In this first case (see Eq. 2) the first ranked system is included in the sum of log-likelihoods, so index j starts at 1. In the second case (see Eq. 3) the sum is formed over

the log-likelihoods of the competing N-best ranked writers only, thus index j starts at 2. In either case an input text line is only assigned to a certain writer if its confidence measure is above a certain threshold. Otherwise, no decision about the identity of the text line is made.

4.2. Writer Verification

For the task of writer verification, the system must decide, based on a verification criterion, whether a text line with a claimed identity is in fact from this writer or whether it is an impostor attempt. The verification criterion used by our system is based on the following confidence measure:

$$cm_{text\ line} = \frac{l_{claimed\ identity} - l_{avg}}{text\ line\ length} \tag{4}$$

where l_{avg} is either given by Eq. 2 or

$$l_{avg} = \frac{1}{N} \sum_{j=1 \wedge j \neq r(t)}^{N+1} l_j \tag{5}$$

The confidence measure in Eq. 4 is calculated from the difference of the log-likelihood score of the claimed identity and l_{avg}, and is normalized by the length of the text line. Similarly to the confidence measure used for writer identification, we can differentiate between calculating l_{avg} based on the score of the N-best ranked writers (see Eq. 2) or based on the N-best ranked competing writers (see Eq. 5, where $r(t)$ is the rank of the claimed identity of text line t). Using these confidence measures, we define the following verification criterion: if the confidence measure is above a certain threshold, we assume that the text line is in fact from the claimed writer; otherwise the input is classified as not being of the claimed identity.

5. Experimental Evaluation

5.1. Database

Our experiments are based on pages of handwritten text acquired in the IAM database [12][1].The database currently contains over 1,500 pages of hand written text from over 500 different writers. Each page contains between five and eleven text lines. For each writer we use five pages of text from which between 27 and 54 text lines are extracted.

To train the writer identification and verification system, we have used 4,307 text lines from 100 different writers. For each writer, the set of available text lines is split into four disjoint subsets, which enables us to perform full-fourfold cross validation experiments. Iteratively, three out of the four sets are used to train the system and the remaining set is used to test the performance of the system. Using cross validation guarantees that the training set does not appear in the test set and that our experiments are text independent.

[1]The database is publicly available at: www.iam.unibe.ch/˜fki/iamDB

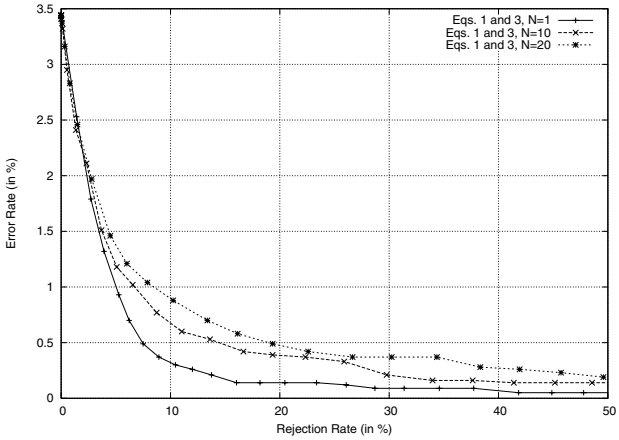

Figure 2. Error-Rejection curve for the identification experiment using different confidence measures.

Furthermore, for the task of writer verification, we extracted an additional 626 text lines, coming from 20 writers, from the IAM database. The writers of these text lines are disjoint from the 100 writers who produced the data set described in the previous paragraph. Consequently, no HMM recognizer exists that was trained on the handwriting of any of these 20 writers. These text lines are used to simulate impostor attempts. The impostor text lines are presented, together with a claimed identity, to the system to test whether it is capable to correctly reject them.

5.2. Writer Identification Experiments

The first set of experiments addresses the problem of writer identification. Using the method described in Section 4.1, we have correctly identified the writer in 96.56% of all cases. This result compares very well to the 94.47% writer identification rate we have achieved in a previous experiment using the same system on 50 writers [20].

We have also conducted a series of experiments using different confidence measures to reject an input in case of uncertainty and calculated the corresponding error-rejection curves (see Fig. 2). The best error rejection curve is achieved using the confidence measure given in Eq. 1 with either Eq. 2 or Eq. 3 and $N = 1$ (Eq. 2 and 3 give identical results). In these cases, by rejecting 5% of the text lines with the lowest confidence score, the error rate drops below 1%. If the rejection rate is further increased to 10%, then the

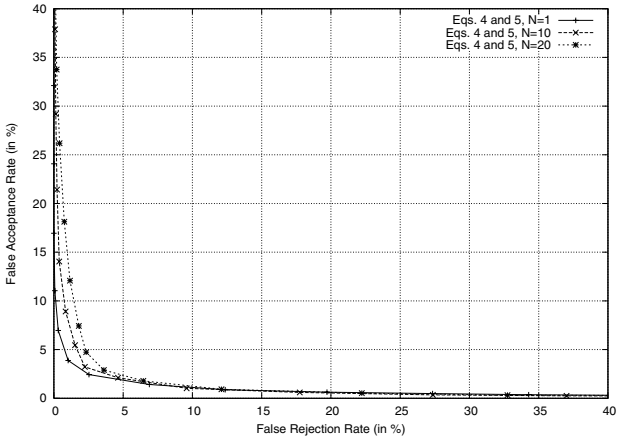

Figure 3. ROC curves of the verification experiment using different confidence measures.

error rate gets as low as 0.32%. Additionally, from Fig. 2 one can observe that increasing the value of parameter N in Eqs. 2 and 3 produces inferior error-rejection curves. For $N = 10$ and $N = 20$ the results obtained under Eq. 2 and 3 are no longer identical, but very similar. For this reason, only Eq. 3 is used in Fig. 2.

5.3. Writer Verification Experiments

The second set of experiments addresses the problem of writer verification. The setting simulates a scenario where there are clients who are authorized to access a system as well as impostors who maliciously try to gain access to it. The test set is formed from two sets. The first set is the set of clients. It consists of 4,307 text lines coming from the 100 writers the system was trained on with their correct identity. The second set is the set of impostors. It is formed of 626 text lines of the 20 writers unknown to the system, each presented seven times with one of the identities of the writers known to the system. Thus 626×7=4,382 text lines with false identities are used. Consequently the complete test set consists of 8,689 text lines whereof about one half has to be accepted and the rest has to be rejected by the verification system.

The results of the verification experiments are given in Fig. 3 where the Receiver Operator Characteristic (ROC) curves [4] for different confidence measures are plotted. Our system performs very well at rejecting impostors as well as at accepting clients. The best ROC curve is produced using the confidence measure based on Eqs. 4 and 5 with $N = 1$. An Equal Error Rate (EER) of about 2.5% is achieved. A False Acceptance Rate (FAR) smaller than 1% is obtained at a False Rejection Rate (FRR) of 16%, and conversely, at a FRR of 1% the FAR is 16%. Similarly to the results of the identification experiment, one can see that increasing the number of writers N to calculate the confidence measures impairs the performance of the system. Replacing Eq. 5 by Eq. 2 gave almost identical results and is omitted in Fig. 3.

6. Conclusions

In this paper, we have presented a system that uses HMM based text line recognizers for the task of text independent off-line writer identification and verification. The basic input units presented to the system are handwritten text lines. From each text line, nine features are extracted. Using these features we train a recognizer and present unknown input text lines to each of the recognizer. As output, each recognizer produces a transcription of the input text line with a log-likelihood score. Based on these scores a ranking in descending order is generated. To identify the author of a text line, we simply choose the first ranked author and assign the text line to it. Using this procedure, a writer identification rate of 96.56% is achieved in a 100 writer experiment. Compared to our previous work [20], we have increased the number of writers by a factor of two from 50 to 100 and our result compares favorably with the 94.47% recognition rate presented there. This is an indication that the proposed approach scales well with an increasing number of writers. Experimenting with a set of confidence measures we can show that by rejecting 5% of the text lines the error rate drops below 1% and by rejecting 10% a recognition rate of 99.68% is achieved.

To verify whether or not a text line is from a claimed author, we use a set of confidence measures to establish a verification criterion. The confidence measures are calculated based on the difference of the log-likelihood of the claimed identity and the average of the log-likelihoods of the N-best ranked or the N-best ranked competing writers, respectively. For different values of parameter N, we have tested our system with a total of 8,689 text lines coming from 100 clients and 20 impostors. Our system performs very well on both tasks of accepting clients and rejecting impostors. An Equal Error Rate (EER) of about 2.5% is achieved.

Currently, we have tested our system on the verification task using unskilled forgeries only. In future work, we plan to address skilled forgeries as well. Additionally, our present verification approach uses all recognizers to check whether a claimed identity is true or not. It would be computationally less expensive to base the decision solely on the system that corresponds to the claimed identity. For such an approach, different rejection strategies and decision criteria are needed. These issues are left for future research.

7. Acknowledgments

This research is supported by the Swiss National Science Foundation NCCR program "Interactive Multimodal Information Management (IM2)" in the Individual Project "Access and Content Protection (ACP)". The authors would like to thank Dr. Simon Günter for fruitful discussions.

References

[1] A. M. Ariyaeeinia and P. Sivakumaran. Analysis and comparison of score normalization methods for text-dependent speaker verification. *Fifth European Conf. on Speech Communication and Technology*, pages 1379–1382, 1997.

[2] A. Bensefia, A. Nosary, T. Paquet, and L. Heutte. Writer identification by writer's invariants. In *Int. Workshop on Frontiers in Handwriting Recognition*, pages 274–279, 2002.

[3] A. Bensefia, T. Paquet, and L. Heutte. Information retrieval based writer identification. In *Seventh Int. Conf. on Document Analysis and Recognition*, pages 946–950, 2003.

[4] F. Bimbot and G. Chollet. Assessement of speaker verification systems. In D. Gibbon, R. Moore, and R. Winski, editors, *Handbook of Standards and Resources for Spoken Language Systems*. Mouton de Gruyter, 1997.

[5] M. Bulacu, L. Schomaker, and L. Vuurpijl. Writer identification using edge-based directional features. In *Seventh Int. Conf. on Document Analysis and Recognition*, pages 937–941, 2003.

[6] H. Bunke. Recognition of cursive roman handwriting – past, present and future. *Seventh Int. Conf. on Document Analysis and Recognition*, pages 448–461, 2003.

[7] S.-H. Cha and S. Srihari. Multiple feature integration for writer verification. In L. Schomaker and L. Vuurpijl, editors, *Proc. Seventh Int. Workshop on Frontiers in Handwriting Recognition*, pages 333–342, 2000.

[8] C. Hertel and H. Bunke. A set of novel features for writer identification. In J. Kittler and M. Nixon, editors, *Audio-and Video-Based Biometric Person Authentication*, pages 679–687, 2003.

[9] F. Leclerc and R. Plamondon. Automatic signature verification: The state of the art 1989–1993. In R. Plamondon, editor, *Progress in Automatic Signature Verification*, pages 13–19. World Scientific Publ. Co., 1994.

[10] G. Leedham and S. Chachra. Writer identification using innovative binarised features of handwritten numerals. In *Seventh Int. Conf. on Document Analysis and Recognition*, pages 413–417, 2003.

[11] U.-V. Marti and H. Bunke. Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system. *Int. Journal of Pattern Recognition and Artificial Intelligence*, 15:65–90, 2001.

[12] U.-V. Marti and H. Bunke. The IAM–database: An English sentence database for off-line handwriting recognition. *Int. Journal of Document Analysis and Recognition*, 5:39–46, 2002.

[13] S. Marukatat, T. Artières, P. Gallinari, and B. Dorizzi. Rejection measures for handwriting sentence recognition. In *Proc. of the Eighth Int. Conf. on Frontiers in Handwriting Recognition*, pages 25–29, 2002.

[14] A. Nosary, L. Heutte, T. Paquet, and Y. Lecourtier. Defining writer's invariants to adapt the recognition task. In *Fifth Int. Conf. on Document Analysis and Recognition*, pages 765–768, 1999.

[15] R. Plamondon and G. Lorette. Automatic signature verification and writer identification – the state of the art. In *Pattern Recognition*, volume 22, pages 107–131, 1989.

[16] R. Plamondon and S. Srihari. On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22:63–84, 2000.

[17] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–285, 1989.

[18] A. E. Rosenberg, J. Delong, C. H. Huang, and F. K. Soong. The use of cohort normalized scores for speaker verification. *Proceeding Int. Conf. on Spoken Language Processing*, pages 599–602, 1992.

[19] H. E. S. Said, T. Tan, and K. Baker. Personal identification based on handwriting. *Pattern Recognition*, 33:149–160, Jan. 2000.

[20] A. Schlapbach and H. Bunke. Off-line handwriting identification using HMM based recognizers. *Accepted for publication in Proc. 17th Int. Conf. on Pattern Recognition*, 2004.

[21] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK book*. 2002.

[22] E. N. Zois and V. Anastassopoulos. Morphological waveform coding for writer identification. *Pattern Recognition*, 33:385–398, Mar. 2000.