Writer Identification Using an HMM-Based Handwriting Recognition System: To Normalize the Input or Not?

Andreas SCHLAPBACH [a] and Horst BUNKE [a]

[a] *Department of Computer Science, University of Bern*
Neubrückstrasse 10, CH-3012 Bern, Switzerland
{schlpbch, bunke}@iam.unibe.ch

Abstract. A system for writer identification based on handwritten text lines is described in this paper. The system uses Hidden Markov Model based recognizers which are designed for text line recognition. Features are extracted from a text line and used to train the recognizers. Prior to feature extraction, normalization operations are applied to a text line. On the one hand, there exists a strong correlation between the text recognition and the writer identification rate, and applying normalization operations increases the text recognition rate. On the other hand, normalization also removes writer-specific information from a handwritten text line. Hence there is a trade-off between optimizing the text recognition performance of our system and keeping writer specific features. In this paper, we study the effect of normalization operations, such as slant correction, width normalization, and vertical scaling, on the identification rate of our system.

1. Introduction

Significant progress has been achieved in handwritten text recognition in recent years (Plamondon and Srihari, 2000; Steinherz, Rivlin and Intrator, 1999; Vinciarelli, 2002). Hidden Markov Models (HMMs) (Rabiner, 1989) have become a powerful tool for handwritten text recognition. HMM-based recognizers have a number of advantages over other approaches (Bunke, 2003). First, they are resistant to noise and can cope with shape variations. Second, they allow to model characters of variable width occurring in the text. Third, HMM-based recognizers are able to implicitly segment a text line into words and characters, a task that is difficult to perform explicitly (Steinherz, Rivlin and Intrator, 1999). Last, there exist standard algorithms for training and testing (Rabiner, 1989).

In this paper, we introduce a system for writer identification to address the problem of determining the author of a sample of handwriting from a set of writers. This system uses an HMM-based handwriting recognizer. Surveys covering work in automatic writer identification and signature verification until 1993 are given in (Leclerc and Plamondon, 1994; Plamondon and Lorette, 1989). Recently, a number of new approaches to writer identification have been proposed. Said et al. (Said, Tan and Baker, 2000) treat the writer identification task as a texture analysis problem using multi-channel Gabor filtering and grey-scale co-occurrence matrix techniques. Srihari et al. (Cha and Srihari, 2000; Zhang, Srihari and Lee, 2003) address the problem of writer verification by casting it as a classification problem with two classes, *authorship* and *non-authorship*. Zois et al. (Zois and Anastassopoulos, 2000) base their approach on single words by morphologically processing horizontal projection profiles. Hertel et al. (Hertel and Bunke, 2003) describe a system for writer identification that extracts a set of features from a text line and uses a k-NN classifier to determine the author. Edge-based directional probability distributions and connected-component contours as features for the writer identification task are proposed in (Bulacu, Schomaker and Vuurpijl, 2003; Schomaker and Bulacu, 2004; Schomaker, Bulacu and Franke, 2004). Bensefia et al. introduce graphemes as features for describing the individual properties of handwriting (Bensefia, Paquet and Heutte, 2003, 2004). Leedham et al. (Leedham and Chachra, 2003) present a set of eleven features which can be extracted easily and used for the identification and verification of documents containing handwritten digits.

The building blocks of our writer identification system (Schlapbach and Bunke, 2004*a*,*b*) are HMM-based recognizers that have been designed and optimized for the task of handwritten text recognition (Marti and Bunke, 2001). The basic idea of our system is to train for each writer a recognizer with data coming from that writer only. Each recognizer thereby becomes an expert on the handwriting of one writer. Confronted with a text line of unknown origin, we expect that the recognizer which was trained with data from this author achieves the highest text recognition rate.

To achieve high text recognition rates the recognizers perform a number of normalization operations on a handwritten text line prior to feature extraction. The result of these operations is a normalized text line where inter- and intra-writer variations are reduced. However, if we remove too much individuality from a handwriting we loose information that is potentially valuable for the task of writer identification. Therefore, there is a trade-off between achieving high text recognition rates and achieving high writer

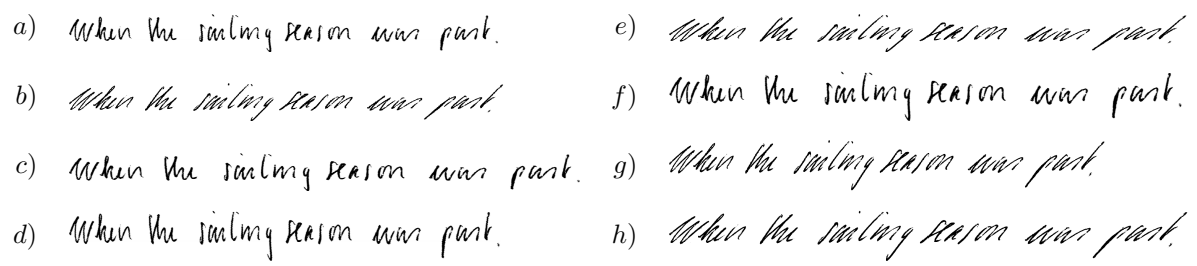

Figure 1. Different normalization operations applied to a text line: a) all three normalization operations applied; b) width normalization and vertical scaling; c) slant correction and vertical scaling; d) slant correction and width normalization; e) vertical scaling; f) slant correction; g) width normalization; h) original text line.

identification rates. The aim of this paper is to measure the effect of three common normalization operations on the writer identification rate of our system. The operations considered are slant correction, width normalization, and vertical scaling.

The remainder of the paper is structured as follows. In the next section, we briefly introduce our system for writer identification using HMM-based recognizers. In Section 3 the normalization operations performed by the recognizers during preprocessing are introduced in detail. The experimental setup and the results are presented in Section 4. Section 5 concludes the paper.

2. Writer Identification Using HMM Based Recognizers

The writer identification system developed previously (Schlapbach and Bunke, 2004*a*,*b*) uses HMM based recognizers that are designed and optimized for the task of handwritten text line recognition (Marti and Bunke, 2001). The basic idea is to use HMM-based handwriting recognizers for the purpose of writer identification. 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 because they are trained on different data each. Intuitively, each HMM can be understood as an expert specialized in recognizing the handwriting of one particular person.

Through a number of normalization operations, the text lines presented to the recognizers are preprocessed. These normalization operations are discussed in detail in the next section. They are applied in order to achieve a higher text recognition rate. After normalization a sliding window moves from left to right over the normalized text lines and extracts nine geometrical 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.

For each upper and lower case character an individual HMM is built. Additionally, we model frequent punctuation marks, such as space, colon, semi-colon, and full stop. Other, infrequent punctuation marks are mapped to a special garbage model because even in large training sets there is not enough data to train the models. Each character HMM consists of 14 states connected in a linear topology. The character models are concatenated to word models which in turn are concatenated to model a complete text line.

For each writer, a text line recognizer is built and trained with text lines of this writer only. In this way we obtain, for each writer, one recognizer that is specially adapted to the individual handwriting style of that writer. A text line that is to be classified is presented to the HMM of each writer. Each HMM outputs a transcription of the input text line together with its log-likelihood score. All log-likelihood scores are sorted in a descending order. A confidence measure (Marukatat et al., 2002) enables us to judge the reliability of the recognition results. If the confidence measure is above a certain threshold, we assign the text line to the highest ranked writer, otherwise the text line is rejected.

The system has been implemented using the HTK toolkit (Young et al., 2002), originally developed for speech recognition. The toolkit employs the Baum-Welch algorithm for training and the Viterbi algorithm for recognition (Rabiner, 1989).

3. Normalization Operations

Before feature extraction, a number of normalization operations are applied to a text line. The following three normalization operations are studied in this paper: *slant correction*, *width normalization*, and *vertical scaling*. It was pointed out in (Marti and Bunke, 2001) that these operations are important to achieve high text recognition rates. Because a high text recognition rate implies a high writer identification rate

(Schlapbach and Bunke, 2004*a*,*b*), normalization is expected to be beneficial not only for text recognition, but also for writer identification. On the other hand, it can be argued that normalization is potentially harmful to writer identification, because the slant, the character width and the proportions between ascender, middle, and descender zones of a text line contain writer specific information. The results of our study on how text line normalization effects the performance in writer identification is presented in the next section. In the remainder of the current section, we provide a brief characterization of the three normalization operations considered in our study. For a detailed description see (Marti and Bunke, 2001).

The first normalization operation applied to a text line is slant correction. Its goal is to bring the handwriting into an upright position (Caesar, Gloger and Mandler, 1993). To correct the slant, the angle between the actual, quasi-vertical strokes and the y-axis has to be known. To calculate this angle, the angle distribution of the writing's contour points is accumulated in an angle histogram. The maximum value of the histogram is the slant angle. This angle is then used to normalize the slant of a text line. An example is presented in Fig. 1. The original text line is depicted in Fig. 1.*h*. Slant correction yields the text line shown in Fig. 1.*f*.

The next operation is width normalization. The goal of this operation is to scale the width of a text line so that the average character length is normalized. This operation is useful because we use a fixed number of states in each character HMM. To determine the average length of a character, the number of black to white transitions in horizontal direction is counted for each text line. The measured value is set in relation to a reference value which is determined empirically over all text lines. The ratio between the calculated value and the reference value gives the scaling factor for the horizontal direction. An example of a width normalized text line is shown in Fig. 1.*g*.

The third normalization operation is vertical scaling. A text line can be divided into three zones: a zone containing the ascenders, the middle zone, and a zone containing the descenders. Our goal is to normalize the height of each of these three zones to a fixed size. This normalization is important in order to reduce the variability of the features used in the HMMs. To actually perform this operation, the upper and lower baselines of the writing have to be determined. To find the lower and upper baseline, the histogram of the horizontal projection of the image of the actual text line is used. The real histogram is matched with an ideal histogram. By means of this matching operation, the location of the upper and lower baseline are detected and the three main writing zones are found. Each of these three zones is then individually positioned and scaled to a predefined height. Application of this normalization operation transforms the text line shown in Fig. 1.*h* into Fig. 1.*e*.

4. Experiments

4.1 Setup

To measure the effect of the normalization operations described in the previous section, we alternately switch them on and off to measure their influence on the writer identification rate. Thus, for the three different normalization operations, we have eight different experimental setups. An example of the data used in each experimental setup is shown in Fig. 1.

Our experiments are based on pages of handwritten text from the IAM database (Marti and Bunke, 2002) [1] . The database currently contains over 13,000 pages of handwritten text from over 650 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. In this paper, we use a subset of the database that is identical to the data set used for writer identification and verification presented in (Schlapbach and Bunke, 2004*b*). In total, we have used 4,307 text lines from 100 different writers including 20,315 words and 5,645 word classes.

For each writer, the set of available text lines is split into four disjoint subsets to perform full four-fold cross validation experiments. Iteratively, three out of the four sets are used to train the system and the remaining set is used to test its performance. This procedure guarantees that text lines in the training set do not appear in the test set and that our experiments are completely text independent.

4.2 Results

In Table 1 the writer identification rates for our eight experimental settings are shown. The first column denotes the experiment, the next three columns indicate whether or not a particular normalization operation was applied. The writer identification rate is given in the fifth column.

In experiment *Exp. 1* all three normalization operations are applied. In this default setup, a writer identification rate of 93.13% is achieved. The best writer identification rate of 97.70% is obtained when the slant correction and width normalization are not applied (*Exp. 5*). This result is significantly better

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

Table 1. Writer identification and text identification rates for different normalization operations. Rows 1 to 8 correspond to rows a to h in Fig. 1.

Exp.	Slant	Width	Vert.	Id. Rate	Same	Other	Δ
1	Yes	Yes	Yes	93.13%	67.98%	15.66%	52.32%
2	No	Yes	Yes	**97.42%**	65.33%	10.88%	**54.45%**
3	Yes	No	Yes	85.19%	63.14%	12.78%	50.36%
4	Yes	Yes	No	92.62%	68.30%	15.43%	52.87%
5	No	No	Yes	**97.70%**	65.56%	10.80%	**54.76%**
6	Yes	No	No	85.35%	62.52%	12.46%	50.06%
7	No	Yes	No	**97.37%**	65.55%	10.80%	**54.75%**
8	No	No	No	75.99%	60.28%	9.49%	50.79%

(at the statistical significance level of 99%) than the default setup. The second best writer identification rate of 97.42% (*Exp. 2*) is achieved when slant is not corrected. The third best writer identification rate of 97.37% results when both slant correction as well as vertical scaling are omitted (*Exp. 7*). The improvements achieved in *Exp. 2* and *Exp. 7* over *Exp. 1* are also statistically significant. In all other setups, the writer identification rates are lower than in the default setup.

Our system uses HMM based handwriting recognizers which are optimized for text recognition. The whole approach is based on the hypothesis that the word recognition rate is higher when the input comes from the writer whose data are used to train the system. We expect a large difference in text recognition performance between the case where the training and the input data come from the same writer and the case where they come from different writers. In columns six and seven of Table 1 word recognition rates are presented. We distinguish between the case where training and input data come from the same writer (column *Same*) and the case where they come from different writers (column *Other*). Obviously, there is a huge difference in word recognition performance between the two cases in each of the eight experimental setups, which very strongly confirms our hypothesis.

In the default setup, the difference between the text recognition rates for the two cases *Same* and *Other* is 52.32%. However, the highest differences Δ between the two cases occur in the three experimental setups where the highest writer identification rates are obtained. The highest difference Δ is observed in *Exp. 5* (54.76%) where both slant correction as well as width normalization are omitted, followed by *Exp. 7* (54.75%) where neither the slant nor the vertical scaling is corrected, and *Exp. 2* (54.45%) where no slant correction is applied. These findings lead to the conclusion that not applying certain normalization operations – in our case slant correction, or slant correction in conjunction with width normalization or vertical scaling – leads to better discrimination between different writers. Although, due to "imperfect" normalization, the word recognition rate generally drops in these cases when compared to *Exp. 1*, the writer identification rates are improved.

5. Conclusion

In this paper, we present a writer identification system that uses HMM based recognizers. For each writer, we train a recognizer and present unknown input text lines to each recognition system. As output we get, for each recognizer, a recognition score for the input text line. Based on these scores a ranking in descending order is generated. If the confidence score is above a certain threshold, we choose the first ranked author and assign the text line to him or her.

Prior to feature extraction, normalization operations are applied to a text line. Three common normalization operations are used and their effect on the writer identification rate is studied. The three operations are slant correction, width normalization, and vertical scaling. Experiments show that if slant correction, or slant correction combined with width normalization or vertical scaling, are omitted then the word recognition rate drops but the writer identification rate can be significantly increased over the default setup where all three normalization operations are applied.

It would be interesting to see the effect of other common normalization operations, such as smoothing or line thinning, on the writer identification and text recognition rates. These normalization operations are not applied in our current recognizer though. Furthermore, the features used by our recognizers are optimized for text recognition but not for writer identification. Using an "optimal" set of features for writer identification should further increase the writer identification rate. The investigation of these issues is left to future research.

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)".

References

Bensefia, Ameur, Thierry Paquet and Laurent Heutte. 2003. Information Retrieval Based Writer Identification. In *Proc. Seventh Int. Conf. on Document Analysis and Recognition*. pp. 946–950.

Bensefia, Ameur, Thierry Paquet and Laurent Heutte. 2004. Handwriting Analysis for Writer Verification. In *Proc. Ninth Int. Workshop on Frontiers in Handwriting Recognition*. pp. 196–201.

Bulacu, Marius, Lambert Schomaker and Louis Vuurpijl. 2003. Writer Identification Using Edge-Based Directional Features. In *Proc. Seventh Int. Conf. on Document Analysis and Recognition*. pp. 937–941.

Bunke, Horst. 2003. "Recognition of Cursive Roman Handwriting – Past, Present and Future." *Proc. Seventh Int. Conf. on Document Analysis and Recognition* pp. 448–461.

Caesar, T., J. M. Gloger and E. Mandler. 1993. Preprocessing and Feature Extraction for a Handwriting Recognition System. In *Proc. Second Int. Conf. on Document Analysis and Recognition*. pp. 408–411.

Cha, S.-H. and S. Srihari. 2000. Multiple feature integration for writer verification. In *Proc. Seventh Int. Workshop on Frontiers in Handwriting Recognition*. pp. 333–342.

Hertel, Caroline and Horst Bunke. 2003. A set of novel features for writer identification. In *Audio-and Video-Based Biometric Person Authentication*, ed. J. Kittler and M.S. Nixon. pp. 679–687.

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

Leedham, Graham and Sumit Chachra. 2003. Writer Identification using Innovative Binarised Features of Handwritten Numerals. In *Proc. Seventh Int. Conf. on Document Analysis and Recognition*. pp. 413–417.

Marti, Urs-Viktor and Horst Bunke. 2001. "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.

Marti, Urs-Viktor and Horst Bunke. 2002. "The IAM–database: An English sentence database for off-line handwriting recognition." *Int. Journal of Document Analysis and Recognition* 5:39–46.

Marukatat, S., T. Artières, P. Gallinari and B. Dorizzi. 2002. Rejection Measures for Handwriting Sentence Recognition. In *Proc. Eighth Int. Conf. on Frontiers in Handwriting Recognition*. pp. 25–29.

Plamondon, R. and G. Lorette. 1989. Automatic signature verification and writer identification – the state of the art. In *Pattern Recognition*. Vol. 22 pp. 107–131.

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

Rabiner, Lawrence R. 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In *Proc. of the IEEE*. Vol. 77 pp. 257–285.

Said, H. E. S., T.N. Tan and K.D. Baker. 2000. "Personal identification based on handwriting." *Pattern Recognition* 33:149–160.

Schlapbach, A. and H. Bunke. 2004*a*. Off-line Handwriting Identification Using HMM Based Recognizers. In *Proc. Seventeenth Int. Conf. on Pattern Recognition*. Vol. 2 pp. 654–658.

Schlapbach, Andreas and Horst Bunke. 2004*b*. Using HMM Based Recognizers for Writer Identification and Verification. In *Proc. Ninth Int. Workshop on Frontiers in Handwriting Recognition*. pp. 167–172.

Schomaker, L. and M. Bulacu. 2004. "Automatic Writer Identification Using Connected-Component Contours and Edge-Based Featurs of Uppercase Western Script." *IEEE Trans. on Pattern Analysis and Machine Intelligence* 26:787–798.

Schomaker, L., M. Bulacu and K. Franke. 2004. Automatic Writer Identification Using Fragmented Connected-Component Contours. In *Proc. Ninth Int. Workshop on Frontiers in Handwriting Recognition*. pp. 185–190.

Steinherz, Tal, Ehud Rivlin and Nathan Intrator. 1999. "Off-line Cursive Script Word Recognition – a Survey." *Int. Journal on Document Analysis and Recognition* 2:90–110.

Vinciarelli, Alexandro. 2002. "A survey of off-line cursive word recognition." *Pattern Recognition* 35:1433–1446.

Young, Steve, Gunnar Evermann, Dan Kershaw, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, Valtcho Valtchev and Phil Woodland. 2002. *The HTK Book*.

Zhang, Bin, Sargur N. Srihari and Sangjik Lee. 2003. Individuality of Handwritten Characters. In *Proc. Seventh Int. Conf. on Document Analysis and Recognition*. Vol. 7 pp. 1086–1090.

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