A Systematic Comparison Between On-Line and Off-Line Methods for Signature Verification with Hidden Markov Models

G. Rigoll, A. Kosmala
Gerhard-Mercator-University Duisburg
Faculty of Electrical Engineering, Dept. of Computer Science
D-47057 Duisburg, Germany
{rigoll,kosmala}@fb9-ti.uni-duisburg.de

Abstract

This paper presents an extensive investigation of various HMM-based techniques for signature verification. Different feature extraction methods and HMM topologies are compared in order to obtain an optimized high performance signature verification system. Furthermore, this paper may be the first systematic comparison of on-line and off-line methods for signature verification using exactly the same database, and leading to the surprising result that the difference in performance for both approaches is relatively small.

1. Introduction

Signature verification is not only a popular research area in the field of pattern recognition and document processing, but also plays an important role in many applications concerned e.g. with security, access control, or financial and contractual matters. For signature verification, it is obvious that signal warping techniques play an important role in order to be able to compare signatures with variations in length and hight, which occurs even for the repetition of the signature of one single person. Due to the importance of the warping problem in signature verification as well as in handwriting recognition applications, the use of HMMs is becoming more and more popular in both areas. HMMs are finite stochastic automata and represent probably the most powerful tool for modeling time-varying dynamic patterns. A good introduction to the basic principles of HMMs can be found in [3]. Applications of HMMs in handwriting recognition can be found in [1,2,4], demonstrating the rising popularity of this technology in handwriting and document analysis. This suggests the idea of using an HMM for training the characteristic signature cues of an individual, and to use this HMM for the verification of a given signature by computing and evaluating the probability that this signature has been generated by the corresponding HMM. In the following chapters we describe the details of our verification algorithms and feature extraction methods and we present results for on-line and off-line verification experiments.

2. Signature Verification Task

A database of 14 persons has been used for the signature verification experiments. Each person delivered 20 signatures, from which 16 examples were taken for training an HMM for each individual. The remaining samples were used for testing. Additionally, 60 forged signatures were used for testing the rejection capabilities of the system. 40 of these samples were so-called "skilled forgeries" of the signatures from 10 individuals, where other persons tried to fake the original signature with great effort and care. 20 samples were so-called "simple forgeries", consisting of primitive stroke combinations, attempting to confuse the HMMs by generating a high observation probability despite their simple structure. The goal of the verification task consisted of having as many as possible original signatures accepted while rejecting all forged signatures.

3. Feature Extraction

As might be expected, feature extraction methods for on-line signature verification are very similar to the methods already developed in our on-line handwriting recognition project. Therefore, the same basic features as described in [1,2] are used in our signature verification experiments. This includes the angle between the strokes of two consecutive sample points in the digitized signature as well as the corresponding difference to the previously computed angle. Also, the "sliding bitmap" feature turned out to be useful for signature verification. As discussed in more detail in [2], in this case a window consisting of 30x30 pixels is slided along the pen trajectory and is summarized in the grey values of a 9-dimensional vector serving as additional feature. However, there are several important differences between the feature extraction methods used for signature verification and for on-line handwriting recognition. As opposed to [1,2], a spatial resampling of the data is purposely not performed, so that the difference in writing speed and writing rythm in various partitions of the signature is retained. Additionally, the variations of writing velocity are represented by including the speed in horizontal and vertical direction directly into the feature vector. Also, the vertical motion as a function of advancement in horizontal direction can be considered as characteristic feature for the signing individual. A DFT is taken from each window of size 10 obtained from this signal. The absolute values of the resulting Fourier spectrum form a 10-dimensional vector that describes the frequencies that are characteristic for a certain individual while delivering his signature. The upper part of Fig. 1 shows the first 3 features calculated for on-line signature verification. All together, the following features are computed: The absolute angle ($\sin\alpha$, $\cos\alpha$), the difference angle ($\sin\Delta\alpha$, $\cos\Delta\alpha$), the bitmap ($x_1 ... x_9$), the velocity (v_x, v_y), the acceleration (a_x, a_y) and the frequency vector ($f_1 ... f_{10}$).

As might be expected, the features for off-line verification are much simpler, because all dynamic information incorporated in the on-line features is not available in this case. Therefore, only the pixel image can be evaluated. This is done in the following way: The difference between maximum and minimum coordinate of the signature is denoted as maximum height of the signature. This distance is subdivided into a certain number of squares, typically 6-10. Each square contains then approximately 10x10 pixels, and the grey value for each square can be computed. In this way, the signature is subdivided into a number of columns, where each column is described by a vector containing 6-10 grey values. The number of columns is determined by the length of the signature and the size of the square previously determined by the number of squares in vertical direction. This feature extraction method is shown in the middle part of Fig. 1.

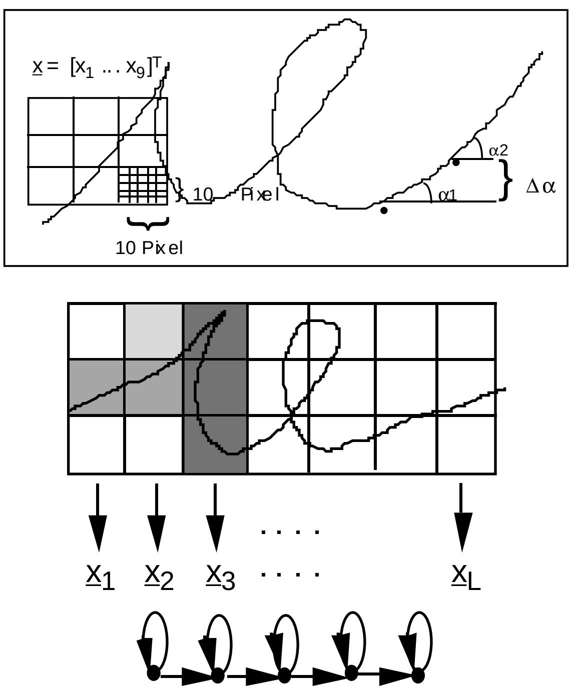

Fig. 1: Feature extraction used for on-line (top) and off-line verification (middle), and HMM topology (bottom).

4. Stochastic Modeling of Signatures with HMMs

Besides the choice of the HMM-topology, the probability density function modeling of the HMMs is the most important part in order to design the most appropriate models for the verification task. We have chosen the technology of discrete HMMs, i.e. the probability density function of a state for emitting a certain feature is approximated by a discrete distribution of vector quantizer indices (see [3] for details). There are two major reasons for having chosen this discrete implementation over the option of continuous modeling using Gaussian mixtures: As shown in [1], discrete models can cope very well with the features typically used in handwriting. Furthermore, a vector quantizer can be derived during training for each individual person. Therefore, the VQ adds one more personal characteristic to the signature verification system. In order to guarantee an efficient VQ design for the large variety of features used in the system, the multiple codebook technique is used, and the feature stream is subdivided into F=7 different streams for angle, Δangle, bitmap, pressure, velocity, acceleration, and Fourier feature, respectively. If for a given HMM-state s the probability of generating the feature f by seeing the i-th label of the f-th VQ-codebook is denoted as $p_s^{(f)}(i_f)$, then the complete state output probability for all seen F features is computed to

$$p_s = \prod_{f=1}^{F} \left\{ p_s^{(f)}(i_f) \right\}^{w_f} \qquad (1)$$

where w_f is a weighting factor used for different weighting of the various streams. For off-line verification, only the bitmap codebook is trained (i.e. F=1). The codebook size for the bitmap feature has been chosen to 100. The codebook size for the on-line features ranges from 2 (for pressure) to 30 (for velocity and angle).

5. Implementation of the Verification Algorithm

As mentioned already previously, the signature of each individual is modeled by one HMM trained with 16 sample signatures of that individual. For the verification of a new signature, the features are extracted as described in the previous section and the Viterbi algorithm (see [3]) is used for computing the probability that the multiple feature stream has been generated by the HMM representing the signing individual. The choice of an individual HMM state number (ranging from 14 to 24) turned out to be very advantageous. The probability obtained from Viterbi decoding was the basis for acceptance or rejection of the signature. Before making the final decision, this value was divided by the difference between the length L of the observation sequence and the number N of HMM states, i.e.

$$p^* = -\frac{\log p}{L - N} \qquad (2)$$

The value N emphasizes the individual signature length: If N is small, one should expect a short signature and in this case, a large L will lead to a large deviation of the value of p^* in comparison to what would have been expected. A signature is accepted if the following condition for the resulting value of p^* is satisfied:

$$\bar{p} - \alpha \cdot \sigma < p^* < \bar{p} + \alpha \cdot \sigma \qquad (3)$$

where $\bar{p}$ is the expected value for p^* and σ is its deviation scaled by a constant factor α. It should be noted that $\bar{p}$ and σ are person-specific values obtained for each individual's HMM during the training phase.

6. Results for On-Line and Off-Line Verification

For signature verification, several error rates are of importance. The False Acceptance Rate indicates the percentage of forged signatures that have been accepted. The False Rejection Rate determines the percentage of valid signatures which have been rejected. If one evaluates the number of correctly accepted original signatures and the number of correctly rejected forgeries, this results in the total correctness which is given in all the following experiments. In one of the first experiments for on-line verification, the information content of the various features has been investigated using each feature separately in a single stream experiment. The result is shown in Fig. 2, which contains several remarkable facts.

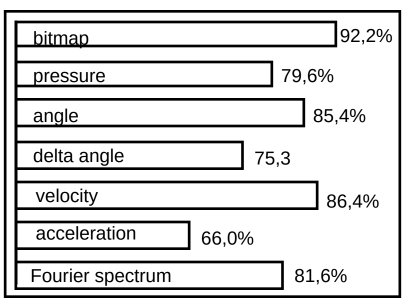

Fig. 2: Verification results for each on-line feature used in single stream HMMs.

It shows the importance of the bitmap feature and the angle feature, which both also serve als major features in handwriting recognition ([2]). The high score for the velocity vector would have been expected, but really surprising is the low importance of the acceleration and the high score for the Fourier feature. Clearly, the combination of several features has to be used in order to obtain high correctness rates. Fig. 3 shows the comparison of several combinations.

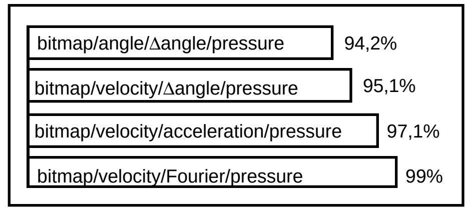

Fig. 3: On-line verification results obtained with different combinations of features.

Fig. 3 shows also the positive influence of the velocity (which seems to be more important than the angle) by comparing case 1 with case 2. It also shows that the acceleration can have a positive effect if used in the right combination with other features (comparison of cases 2 and 3). The best rate with 99% has been obtained with only 4 features (bitmap, velocity, Fourier, pressure).

The best result of 98.1% for off-line verification has been obtained with a feature vector $\underline{x}$ (as shown in the middle part of Fig. 1) of dimension 6 and a codebook size of 30 for that vector. The most remarkable result, however, is the small difference between the best on-line and off-line correctness rates (99.0% vs 98.1%, respectively).

7. Conclusion

This paper has presented an extensive and systematic comparison of several HMM-based techniques for signature verification. The importance of individual features, topologies, and threshold values for each person has been demonstrated. Probably the most important outcome is the high correctness rate for on-line verification, showing the potential of this approach for real-world applications. It is also worthwhile to note that new users can be incorporated easily through the automatic training capabilities of the classifier. A positive surprise is the high rate obtained for off-line recognition. Again, this can be considered as important practical result, because off-line recognition is easier to realize and needs less sensitive equipment.

References

[1] G. Rigoll, A. Kosmala, J. Rottland, Ch. Neukirchen: "A Comparison between Continuous and Discrete Density Hidden Markov Models for Cursive Handwriting Recognition", Proc. IEEE -ICPR, Vienna, 1996, Vol. II, pp. 205-209.

[2] A. Kosmala, J. Rottland, G. Rigoll: "Improved On-Line Handwriting Recognition Using Context-Dependent Hidden Markov Models", Proc. ICDAR, Ulm, 1997, pp. 641-644.

[3] L.R. Rabiner, B.H. Juang: "An Introduction to Hidden Markov Models", IEEE ASSP Magazine, 1986, pp. 4-16.

[4] L. Yang: "Processing and Recognition of Handwriting in Multimedia Environments", PhD Dissertation, Delft University of Technology, The Netherlands, 1995.