

Dynamic Signature Verification Using Discriminative Training

Gregory F. Russell Jianying Hu Alain Biem
IBM T.J. Watson Research Center,
1101 Kitchawan Road, Route 134, Yorktown Heights, NY 10598
{gfr,jyhu,biem}@us.ibm.com

Andre Heilper Dmitry Markman
IBM Haifa Research Lab, University Campus, Carmel Mountains, Haifa, 31905, Israel
{heilper,markman}@il.ibm.com

Abstract

In this paper we describe a new approach to dynamic signature verification using the discriminative training framework. The authentic and forgery samples are represented by two separate Gaussian Mixture models and discriminative training is used to achieve optimal separation between the two models. An enrollment sample clustering and screening procedure is described which improves the robustness of the system. We also introduce a method to estimate and apply subject norms representing the "typical" variation of the subject's signatures. The subject norm functions are parameterized, and the parameters are trained as an integral part of the discriminative training. The system was evaluated using 480 authentic signature samples and 260 skilled forgery samples from 44 accounts and achieved an equal error rate of 2.25%.

1. Introduction

Signature verification is a commonly used biometric authentication method. Compared to other forms of biometric authentication such as finger print or iris verification, signature verification has the advantage that it is an historically well established and well accepted approbation method and is thus perceived to be less intrusive than many modern alternatives. This property makes it particularly attractive to applications in banking, retail and hospitality industries.

Automatic signature verification is divided into two main areas: static (or offline) signature verification where signature samples are scanned into image representations, and dynamic (or online) signature verification (DSV) where signature samples are collected from a digitizing tablet capable of recording pen movements during writing. The lat-

ter is more reliable because the extra information captured reflecting the dynamics of writing makes signatures much more difficult to forge. A recent survey of dynamic signature verification can be found in [7].

In this paper we describe a new dynamic signature verification system using the discriminative training framework. Discriminative training is a method focused on maximizing the separation between the target classes and has been used successfully in speech recognition and handwriting recognition [1], but has not previously been applied to signature verification. An enrollment sample clustering and screening procedure was developed to improve the robustness of the system. Furthermore, we introduce a novel method to estimate the *subject norms* representing the "typical" variation of each subject. The estimation of the parameters used to compute these norms was implemented as an integral part of the discriminative training framework, leading to maximal separation of the genuine and forgery classes across all subjects.

Although many modern dynamic signature collection devices offer extra information such as pressure and pen tilt, only the basic XY coordinates along with the timing information are currently used in our system to make it compatible with the widest possible range of devices.

2. System Overview

Figure 1 shows the main modules of the signature verification system. In all cases, processing begins with low pass filtering and noise filtering, and rescaling. For enrollment, the samples are then screened for possible outliers. For the remaining enrollment samples, called references, Dynamic Time Warping (DTW) is performed to align each pair of references. Dissimilarity measures are then computed for each pair. These reference-reference dissimilarity values are used to estimate the *subject norm* for each dissimilarity measure,

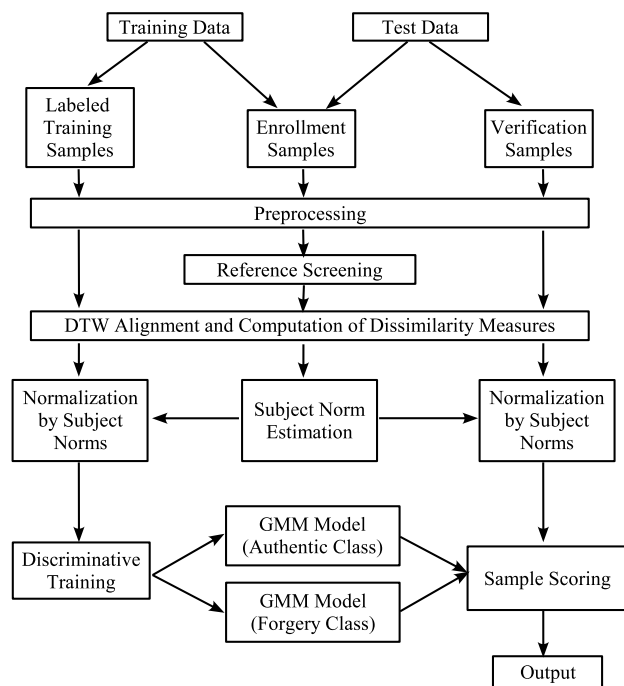


Figure 1. System Diagram.

which represents the "typical" amount of variation among this subject's signatures.

For a verification sample, after filtering and rescaling, some simple global features are computed and compared to those of the references from the claimed subject to screen out obvious outliers. If the sample passes, it is then aligned with each reference from the claimed subject using DTW, and the dissimilarity measures are computed. These measures are then normalized by the subject norms. During training, the normalized dissimilarity vectors along with the forgery/authentic labels are used in a discriminative training procedure to train two Gaussian Mixture models (GMM), one for the authentic class and one for the forgery class.

During testing or system deployment, the normalized dissimilarity vectors are used against the GMMs to evaluate the sample's likelihood of being authentic or forged. The dissimilarity vector computed between the verification sample and each one of the references produces a score which is the difference between the log likelihood of the authentic and forgery models. The final score is the maximum of the scores produced by all references and the verification samples is accepted as authentic if the final score is larger than

the threshold, nominally zero. The threshold can be easily adjusted in different applications to achieve different trade-offs between the false acceptance rate and the false reject rate.

3. Dissimilarity Measures

Two types of dissimilarity measures are used in our system: global measures derived from overall characteristics of signatures and point-wise measures computed after two signature samples are aligned against each other. The former is a coarse-level measure designed to identify obvious forgeries, and latter is used for more detailed analysis on verification samples that pass the coarse-level test.

The global measures used in our system are the differences between the following quantities: number of Y maximums, the ratio between the accumulated shifts along Y and X direction, and number of curvature extrema.

The alignment between two signature samples is computed using DTW with a cost function based on four difference metrics – Euclidean distance, stroke direction, curvature, and curve length to stroke endpoints. These metrics are combined using a sum of sigmoid functions, and the costs are weighted by the length of the segment. This sum of sigmoids avoids the pathological behavior often seen in DTW, without having to use constraints on the alignment.

Three point-based distance measures are computed after DTW alignment. The first is simply the cost computed during DTW alignment, and consequently reflects a rather complex function incorporating location, direction, and curvature of the strokes. The second distance measure, called affine distance, is designed to capture the local shape differences between two signatures. It is defined as the average Euclidean distance between corresponding points, after an optimal affine transform between corresponding sections of the signature. The third is the output of a simple point-wise correlation function, applied to piecewise segments of the acceleration vector [8].

To these three pointwise distance measures, we add a fourth measure to form a four dimensional feature distance vector used in the GM models. The fourth measure is simply the log of the ratio of the pointcount in one signature to the pointcount in the other signature. This measure, unlike the others, may be positive or negative, and is nominally zero.

3.1. Reference Sample Screening

Reference signature samples are used in two ways in our system. First each reference signature is used as a template against which a new signature is compared. Second, all reference samples for each subject are compared against each

other to estimate the typical range of variation for this particular subject. For both purposes, it is important to include in the analysis only "valid" reference samples, i.e., samples that are truly representative of this particular subject's signature. Unfortunately, reference samples collected during enrollment sessions are not always so reliable and often contain "outliers" that can seriously degrade the performance of the system.

We developed a procedure for reference sample screening using hierarchical clustering [4] with a distance measure computed as the L2 norm of the three point-based distance measures. The procedure is based on the observation that valid signature samples are more similar to each other, and thus tend to form a dominant cluster. The outliers can then be identified as samples that are far from the members of the dominant cluster. Hierarchical clustering has been previously investigated for template selection for fingerprint verification [9], however in a different manner: there it was used to select K different clusters to represent the variability of fingerprint samples.

3.2. Normalization Across Subjects

Since signature is a behavioral biometric, the "typical" amount of deviation among genuine signature samples varies widely from subject to subject. Because of this inherent variation, using a straightforward common threshold for all subjects during verification does not work well. Jain et. al. demonstrated that using subject-dependent thresholds could lead to over ten fold improvement in verification accuracy [5]. However, estimating subject dependent thresholds separately for each subject is difficult because there are typically not enough signature samples representing both the genuine and forgery classes for an individual subject.

An alternative to using subject-dependent thresholds is to first normalize the distance between an input signature and a reference signature by a value representing the "typical" distance between reference signatures. This would in effect map the distances to signatures from different subjects to the same space, thus allowing the pooling of training samples from all subjects in order to identify a common boundary separating the genuine and forgery classes.

For each feature distance, we term its "typical" value between reference signatures of a subject the *subject norm*, and experimented with two different ways of estimating this norm.

The first method, called **Median Norm**, uses the median feature distance among the reference signatures in the dominant cluster. Unlike previous comparable methods (e.g., [7]), we use the median instead of average because it is less sensitive to outliers. The impact of potential outliers is further reduced in our system by taking the median over the dominant cluster instead of the whole reference set.

In the second method, called **β norm**, we replace the threshold used in hierarchical clustering with a trainable parameter β . The subject norm \aleph is a function of β and computed as follows. For an individual subject, we define the radius of a cluster, r_k , as the greatest difference between the "central" reference, and all other references in the cluster, using the k th feature distance. We use this definition to hierarchically cluster the subject references with respect to each of the 3 point-based distance measures. We then sort the intra-cluster radii in the complete hierarchical clustering tree in increasing order. For a subject S with M references, $\aleph_{S,k} = F_k(\beta \times (M - 1))$ where function $F(\cdot)$ is a cubic spline that interpolates the M radii.

4. Discriminative Training of Gaussian Mixture Models

The concept of discriminative training has recently been gaining momentum as a more powerful alternative to the standard generative training approach. The generative approach attempt to reduce classification error by finding the best model possible for each class whereas the Discriminative Training methodology focuses on estimation boundaries between classes.

Discriminative training can achieved by embedding the discriminative capabilities within the structure of the recognizer (e.g, SVMs) or by devising a discriminative objective criterion as a basis for classifier optimization. The discriminative training approach used in this paper is derived from the Minimum Classification Error (MCE) criterion [6]. MCE training defines an objective function which is a smooth approximation of the error rate on the training.

4.1. Gaussian Mixture Models

A GMM implements a probability density function as a sum of Normal densities:

$$P(x; \lambda) = \sum_{i=0}^L w_i \mathcal{N}(x; \mu_i, \Sigma_i) \quad (1)$$

where w_i, μ_i, Σ_i are respectively the mixing weights, the means and the covariance matrix of the i -th Normal density with mixing weights w_i summing to one.

Our system uses two GMMs of parameters λ_a and λ_f to represent authentic signatures and forgeries, respectively. Let $P_a(x) = P(x; \lambda_a)$ and $P_f(x) = P(x; \lambda_f)$ be the pdfs of the two GMMs. For each feature-distance vector x , the system generates a score $d(x) = \log P_a(x) - \log P_f(x)$ defined as difference between the authentic log likelihood and the forgery log likelihood. A signature is classified as an authentic whenever the score is positive and as a forgery otherwise.

As explained in section 3.2, each signature is represented by a 4-dimensional normalized feature-distance vector $x = [x_k]$ for $k = \{1, \dots, 4\}$, where each dimension of x corresponds to one of the distance measure described in section 3.2. Each component $x_k = \frac{y_k}{\aleph_k}$; x_k is a normalized version of the original component of sample y_k , where the normalizing factor \aleph_k is the subject norm for the k -th feature distance. As described in section 3.2, \aleph_k is a function of β_k .

In addition, we implemented different class-dependent norms α_a and α_f for the authentic model and the forgery model, respectively. α_a weights all inputs to the authentic models whereas α_f weights inputs to the forgery models. Ideally, α_a reflects the overall characteristics of authentic signatures across all subjects while α_f reflects the overall characteristics of forgeries. We define a subject complexity as measure of the inherent complexity of a subject signatures. The term is computed as function of a subject's complexity measures, namely the median frequency in the x-axis coordinate, the log of the duration of the signature (excluding time between strokes), and the integrated length of the strokes, when normalized to a fixed standard deviation in the y-axis. α_a and α_f are parameterized functions of the all subject norms and complexities with separate trainable parameters.

4.2. Minimum Error training

Let $\lambda = \{\lambda_a, \lambda_f, \beta_k, \alpha_a, \alpha_f\}$ representing the set of all parameters of the system. The minimum-error training process is as follows. First, we define an smooth error function $\ell(\cdot)$ which approximates the step-wise 0-1 error function:

$$\ell(x; \lambda) = \begin{cases} \ell(-d(x)) & \text{if } x \text{ is authentic} \\ \ell(d(x)) & \text{otherwise} \end{cases} \quad (2)$$

where $\ell(\cdot)$ is typically the sigmoid function.

Having defined the MCE loss, the next step is to minimize the objective function which is the average loss $L(\lambda)$:

$$L(\lambda) = \frac{1}{N} \sum_x \ell(x; \lambda). \quad (3)$$

where N is the total number of data. Since $\ell(x; \lambda)$ approximates the 0 – 1 loss function, $L(\lambda)$ has its values close to the empirical error rate, given a body of training data, meaning that minimization of the objective function optimizes, almost directly, the performance of the system and provides a convenient method to monitor both the MCE learning process and the performance of the system as learning proceeds.

We used the Quick-prop algorithm [3], which combines a gradient descent technique and the Newton algorithm and uses an approximation of the Hessian matrix that does not

	Subjects	Authentics	Forgeries
Training Set	226	2933	1661
Validation Set	78	826	372
Test Set	44	480	260

Table 1. Training, validation and test sets.

require any extra computation as follows

$$\lambda_{\tau+1} = \lambda_{\tau} - [\nabla^2 L(\lambda_{\tau}) + \mu I]^{-1} \nabla L(\lambda_{\tau}) \quad (4)$$

where μ is a learning rate. The Hessian matrix is assumed to be diagonal. The Hessian is approximated by means of first order derivatives [2]:

$$\frac{\partial^2 L(\Lambda_{\tau})}{\partial^2 \lambda} \approx \frac{\frac{\partial L(\Lambda_{\tau})}{\partial \lambda} - \frac{\partial L(\Lambda_{\tau-1})}{\partial \lambda}}{\lambda_{\tau} - \lambda_{\tau-1}}. \quad (5)$$

5. Experiments

5.1. Data

The signature databases used for development and testing of the system consist of 3 sets of signatures collected around 1994, mostly in the U.S., and three additional databases collected in 2003 in the U.S. The latter three databases were collected using a Wacom Intuos tablet at 100 samples/sec, .01mm resolution. The earlier signatures were collected using unknown hardware, at 100 samples/sec, but at lower resolution. One of the 1994 data sets contains some Hebrew signatures, all others contain only English signatures. These six databases represent a wide range of hardware configurations and collection environment. For each subject there are six to ten enrollment samples and variable numbers of authentic verification samples and skilled forgery samples.

The databases were split into three data sets as summarized in Table 1: roughly 70% of subjects in each database went into the training set, 20% into the validation set, and 10% into the test set. Note that there is no overlap among the three data sets at the subject level.

5.2. Experimental setup and Results

The GMMs were initially trained with the EM algorithm. After EM initialization, the eigenspaces of the initial Gaussian mixtures were identified by diagonalizing the covariance matrices. The unitary transform required to diagonalize was then used to transform the means and variances (which then become simple vectors), and used to transform the subject normalized data. The GMMs were then represented using orthogonal Gaussians, and for the forgery and authentic models, the 4 means and 4 variances for each

Gaussian, and the mixing weights among the four Gaussians were all made trainable, while the unitary transform matrices were held fixed.

To these 36 parameters for the authentic GMM, and 36 parameters for the forgery GMM, additional 6 parameters for the subject norm functions (separate for authentic and forgery GMMs) are included, for a total of 78 trainable parameters.

Discriminative training was then performed on all trainable parameters. The smooth error function for this phase was a simple sigmoid on the classification scores, with authentic samples positively weighted, and forgeries negatively weighted. Successful forgeries were weighted more heavily than failed authentic samples. Training was stopped when the objective function reached a plateau and the resulting model was evaluated on the test set.

We compared our discriminatively-trained system with an earlier Neural Net (NN) based system. For the NN based system, the same preprocessing and feature extraction modules were used. The feature distance vectors were normalized using the Median Norm method described in Section 3.2, then used to train a NN classifier. Following common practice in NN training, five different classifiers were trained using five different randomly selected sets of initial parameters, and the classifier that produced best result on the validation set were chosen to produce results on the test set.

The NN based system produced an Equal Error Rate (ERR) of 2.7%. The ERR of the GMM models obtained using EM training alone was 5.5%, worse than the NN model. However, after discriminative training on both the Gaussian and normalization parameters, the ERR was dramatically reduced to 2.25%, achieving a 17% relative error reduction over the best NN model.

6. Conclusions

We have demonstrated that the use of discriminative training can improve the performance of a signature verification system. Discriminative training has been widely recognized recently as a more powerful alternative to the standard generative training approach. Although it has been used successfully in speech recognition and handwriting recognition, it had not previously been applied to signature verification. In this paper we have described a novel signature verification system based on the discriminative training concept, where the authentic and forgery samples are represented by two separate Gaussian Mixture models and discriminative training is used to achieve optimal separation between the two models. We also introduced a novel method to estimate the *subject norms* representing the “typical” variation and complexity of each subject. The estimation of the parameters used to compute these norms was im-

plemented as an integral part of the discriminative training framework. The system was evaluated using 480 authentic signature samples and 260 skilled forgery samples from 44 accounts, and achieved a 17% relative reduction in Equal Error Rate compared to an earlier Neural Net based system.

References

- [1] A. Biem. Minimum Classification Error Training of Hidden Markov Model for Handwriting Recognition. In *Proceedings of ICASSP*, volume 3, pages 1529–1532, 2001.
- [2] A. Biem. Minimum Classification Error Training for Online Handwritten Word Recognition. In *Proceedings of IWFHR*, pages 61–66, 2002.
- [3] S. E. Fahlman. An empirical study of learning speed in back-propagation networks. Technical report, Department of Computer Science, Carnegie Mellon University, 1988.
- [4] A. Jain and R. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.
- [5] A. Jain, F. Griess, and S. Connell. On-line signature verification. *Pattern Recognition*.
- [6] S. Katagiri, B.-H. Juang, and C.-H. Lee. Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method. *Proceedings of the IEEE*, 86(11):2345–2373, 1998.
- [7] A. Kholmatov and B. Yanikoglu. Biometric authentication using online signatures. In *Lecture Notes in Computer Science - ISCIS*, Oct. 2004.
- [8] J. D. W. T. K. Worthington, T. J. Chainer and S. C. Gundersen. IBM dynamic signature verification. *Proceedings of the Third IFIP International Conference on Computer Security*, 1985.
- [9] U. Uludag, A. Ross, and A. Jain. Biometric template selection and update: a case study in fingerprints. *Pattern Recognition*, 37:1533–1542, 2004.