IEICE TRANS. INF. & SYST., VOL.E100–D, NO.2 FEBRUARY 2017

PAPER

Text-Independent Online Writer Identification Using Hidden Markov Models

Yabei WU[†a)], *Student Member*, Huanzhang LU[†b)], *and* Zhiyong ZHANG[†c)], *Nonmembers*

SUMMARY In text-independent online writer identification, the Gaussian Mixture Model(GMM) writer model trained with the GMM-Universal Background Model(GMM-UBM) framework has acquired excellent performance. However, the system assumes the items in the observation sequence are independent, which neglects the dynamic information between observations. This work shows that although in the text-independent application, the dynamic information between observations is still important for writer identification. In order to extend the GMM-UBM system to use the dynamic information, the hidden Markov model(HMM) with Gaussian observation model is used to model each writer's handwriting in this paper and a new training schematic is proposed. In particular, the observation model parameters of the writer specific HMM are set with the Gaussian component parameters of the GMM writer model trained with the GMM-UBM framework and the state transition matrix parameters are learned from the writer specific data. Experiments show that incorporating the dynamic information is capable of improving the performance of the GMM-based system and the proposed training method is effective for learning the HMM writer model.

key words: *online handwriting, text-independent writer identification, HMM*

1. Introduction

The identification of a person on the basis of online handwriting data is a useful biometric modality with application in the environment of smart meeting rooms [1], [2] and constitutes an exemplary study area within the research field of behavioral biometrics. In comparison with physiological biometrics (e.g., iris, fingerprint, hand geometry), handwriting is less invasive, but the achievable identification accuracy is less impressive due to the large variability of the behavior-derived biometric templates [3].

Research in online writer identification has received significant interest in recent years due to its potential applicability. Arti Shivram et al. [4] propose a Latent Dirichlet Allocation(LDA)-based approach. The edge-hinge features [3], [5] are used in this system. Marcus Liwicki et al. [1], [2] perform writer identification by modeling each writer's handwriting with a specified GMM. Both the above systems assume that the features extracted at adjacent points are independent. In the related application of online signature verification, HMM has been used to model online handwriting data [6]–[9]. The constraint of left-to-right model is usually imposed to incorporate the prior information that online signature verification is a text-dependent task, i.e. the text content of the query sample is known. To the best of our knowledge, HMM has been not applied in text-independent online writer identification. This is similar with the situation in speaker recognition. In text-dependent applications, where there is strong prior knowledge of the spoken text, additional temporal knowledge can be incorporated by using HMMs. To date, however, use of more complicated likelihood functions, such as those based on HMMs, has shown no advantage over GMMs for text-independent tasks [10].

In this paper, a writer model based on HMM is introduced and evaluated for text-independent online writer identification. Intuitively, the hidden states of HMM represent some general writing style classes. The HMM models not only the distribution of feature vectors in each style, but also the distribution of style transitions using the state transition matrix. The use of HMM for modeling writer identity is motivated by the fact that style transition patterns provide extra information for writer identification. For example, if using the speed feature, two time-adjacent styles will contain the acceleration information. And the state transition matrix of HMM enables the use of these extra information for modeling writer identity.

In text-independent tasks, GMM writer models trained with the GMM-UBM framework are popular and have acquired excellent performance. GMM uses Gaussian distribution to model the features in each writing style as HMM, but does not impose any Markovian constraints, which neglects the dynamic information between writing styles. In order to extend the GMM-UBM system to incorporate the dynamic information between adjacent feature vectors for writer identification, the HMM with Gaussian emission density is used to model writer identity and a new learning method is proposed. The training method is performed by setting the writing style class parameters of HMM with the parameters of GMM writer model trained with the GMM-UBM framework and then learning the style transition distribution parameters from the writer specific handwriting. This method allows a clear investigation of the effect of the temporal modeling between styles.

The proposed HMM writer model is experimentally evaluated on the IAM-OnDB database consists of more than 200 writers' on-line data acquired from a white-board. The performance comparison between the proposed HMM

Manuscript received June 2, 2016.
Manuscript revised September 27, 2016.
Manuscript publicized November 2, 2016.

[†]The authors are with the Automatic Target Recognition Laboratory, National University of Defense Technology, Changsha, China.
a) E-mail: wuyabei1989@yahoo.com
b) E-mail: luhz@nudt.edu.cn
c) E-mail: zzy07@sina.com

DOI: 10.1587/transinf.2016EDP7238

writer models and the GMM writer models illustrates that temporal modeling between the writing style classes is important for text-independent task and further improves the writer identification performance. The performance of HMM with GMM observation model trained using the user-adapted UBM(UA-UBM) and the user-specific HMM(US-HMM) method in [9] is also studied. The results show that the proposed HMM writer model outperforms. As indicated in [11], good initial estimates of the observation model parameters are essential for training the HMM with continuous observations. So the style classes learned with the GMM-UBM framework are also good style classes for HMM writer model and the proposed method provides an effective way to learn HMM writer model.

The rest of the paper is organized as follows. We first present the features used in this work in Sect. 2. The proposed HMM-based system is presented in Sect. 3. Empirical evaluation of our method is examined in Sect. 4, followed by conclusions in Sect. 5.

2. Feature Extraction

Online white-board data are used in this paper. The text written on the white-board is encoded as a sequence of time-stamped (x,y)-coordinates. From this sequence, we extract a sequence of feature vectors and use them to train the classifier. The text is composed of many strokes. A stroke starts with a pen-down movement of the pen and ends with the next pen-up movement. Thus a stroke is a sequence of points during a certain time interval when the pen-tip touches the white-board. Figure 1 illustrates the computing of the features in this paper. Angle θ_i denotes the angle between the horizontal line and the line (p_i, p_{i+1}), and angle ϕ_{i+1} represents the angle between the lines (p_i, p_{i+1}) and (p_{i+1}, p_{i+2}). For a given stroke consisting of points p_1 to p_n, the following features for each point are computed.

- The speed at p_i

$$v_i = \frac{\Delta(p_i, p_{i+1})}{\Delta_t(p_i, p_{i+1})}$$

where $\Delta(p_i, p_{i+1})$ is the distance between points p_i and p_{i+1}, $\Delta_t(p_i, p_{i+1})$ denotes the time between the two points.

- The writing direction at p_i, i.e. the cosine and sine of angle θ_i

$$\cos\theta_i = \frac{\Delta_x(p_i, p_{i+1})}{\Delta(p_i, p_{i+1})}$$

$$\sin\theta_i = \frac{\Delta_y(p_i, p_{i+1})}{\Delta(p_i, p_{i+1})}$$

where Δ_x is the horizontal distance while Δ_y is the vertical distance.

- The curvature at p_i, i.e. the cosine and sine of angle ϕ_i

$$\cos\phi_i = \cos\theta_{i-1}\cos\theta_i + \sin\theta_{i-1}\sin\theta_i$$

$$\sin\phi_i = \cos\theta_{i-1}\sin\theta_i - \sin\theta_{i-1}\cos\theta_i$$

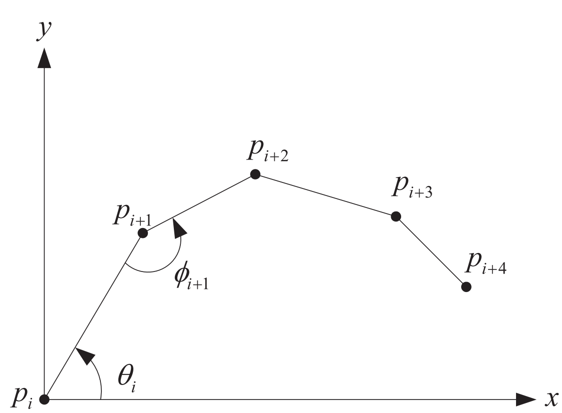


Fig. 1 Illustration of the point-based features.

These features constitutes a 5-dimensional feature vector, which is used in this paper for writer identification. This is just the point-based feature set in [1], [2].

3. HMM Writer Model

Generative classifiers are used commonly in online writer identification, which rely on building a generative model to model the distribution of the features extracted from the handwriting of one person [1], [2], [12]. During the training procedure, we get a model for each writer. In the testing phase, a text of unknown identity is presented to each model. Each model returns a log-likelihood score and the text is assigned to the person whose model produces the highest log-likelihood score. In this work, HMM is used to model the writer's handwriting. But the training method of the HMM is based on GMM trained with the GMM-UBM framework, which is explained in the following section.

3.1 GMM Writer Model

In [2], GMM is used to model distribution of the feature vectors extracted from a person's handwriting. For a D-dimensional feature vector $\mathbf{x}$, the probability density is defined as

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{N} w_i N(\mathbf{x}|\mu_i, \Sigma_i) \tag{1}$$

where the mixture weights w_i sum up to one, N is the number of Gaussian components. $N(\mathbf{x}|\mu_i, \Sigma_i)$ denotes a multivariate normal probability density function with mean μ_i and covariance matrix Σ_i. The parameters of a writer's density model are denoted as $\lambda = \{w_i, \mu_i, \Sigma_i\}, i = 1, 2, \cdots, N$. In the GMM-based system for online handwriting identification, a GMM λ^u is generated for each writer u.

In the GMM-UBM framework, the writer specific GMM is obtained by a two-step training procedure. In the first step, all training data from all writers are used to train a single, writer independent GMM λ^{UBM}

as UBM. The UBM is trained using the expectation-maximization(EM) algorithm, which is denoted by $\lambda^{UBM} = \{w_i^{UBM}, \mu_i^{UBM}, \Sigma_i^{UBM}\}, 1 \leq i \leq N$. In the second step, for each writer, the writer specific model is obtained by adapting the UBM to the training data from that writer using the maximum a posteriori(MAP) adaptation. In this paper, only the adaptation of the means is performed. The specifics of the adaptation are as follows. Given the UBM λ^{UBM} and the training vectors of writer u, $X^u = \{\mathbf{x}^k_{1:T_k}\}, k = 1 : K$, where K denotes the number of strokes in the data set and T_k denotes the number of vectors in the stroke k, we compute the soft count of samples belonging to the Gaussian mixture i

$$T_i = \sum_{k=1}^{K}\sum_{t=1}^{T_k} p(i|\mathbf{x}^k_t) \tag{2}$$

And the maximum likelihood mean for mixture i is computed as

$$\mu_i = \frac{1}{T_i}\sum_{k=1}^{K}\sum_{t=1}^{T_k} p(i|\mathbf{x}^k_t)\mathbf{x}^k_t \tag{3}$$

Then, the MAP mean of mixture i for writer u is computed as

$$\mu_i^u = \frac{T_i}{T_i + T_r}\mu_i + \frac{T_r}{T_i + T_r}\mu_i^{UBM} \tag{4}$$

where T_r is a fixed relevance factor. A more complete MAP formula description for GMM can be found in [10]. The MAP approach is useful for dealing with problems posed by sparse training data of each writer and has acquired improved performance in comparison with the writer specific GMM trained directly with the writer's data.

3.2 Model Interpretations and Learning

An HMM assumes the observation sequence is generated by an underlying discrete stochastic process that is not observable(it is hidden). The joint probability of the hidden state sequence $z_{1:T}$ and observation sequence $\mathbf{x}_{1:T}$ is defined as

$$\begin{aligned} p(z_{1:T}, \mathbf{x}_{1:T}) &= p(z_{1:T})p(\mathbf{x}_{1:T}|z_{1:T}) \\ &= [p(z_1)\prod_{t=2}^{T} p(z_t|z_{t-1})][\prod_{t=1}^{T} p(\mathbf{x}_t|z_t)] \end{aligned} \tag{5}$$

where $z_t \in \{1, 2, \cdots, N\}$, N denotes the number of states. The HMM is characterized by $\lambda = \{\pi, A, B\}$. $\pi = \{\pi_i\}$ denotes the initial state distribution, $\pi_i = p(z_1 = i), 1 \leq i \leq N$. $A = \{a_{ij}\}$ denotes the state transition distribution, $a_{ij} = p(z_t = j|z_{t-1} = i), 1 \leq i, j \leq N$. B denotes the observation model parameters.

The use of HMM for modeling writer identity is motivated by the fact that two time-adjacent feature vectors $(\mathbf{x}_t, \mathbf{x}_{t+1})$ can provide more discriminative information than only one feature vector. For example, the features used in this work are speed, direction and curvature. The co-occurrence of adjacent speeds can provide the acceleration information. The co-occurrence of directions provides the curvature and the co-occurrence of curvatures provides information about the change of curvature. And HMM is capable of modeling writer identity using these dynamic information due to its state transition matrix. Intuitively, the hidden states in HMM represent some general writer-dependent handwriting styles. The HMM models not only the distribution of feature vectors in each style, but also distribution of style transitions using the state transition matrix. Given the HMM writer model, the state transition matrix shows which style transition is preferred by this writer, i.e. which two time-adjacent classes of feature vectors are preferred. And the observation sequence generated by the style sequence containing more high probability style transition patterns is more likely to acquire a high score. So HMM is one way to model the distribution of two time-adjacent feature vectors.

The GMM writer model trained with the GMM-UBM framework has acquired excellent performance. Similarly to HMM writer model, the Gaussian components of GMM can also be interpreted as general handwriting styles. However, GMM models the distribution of the underlying styles, which neglects the dynamic information between styles. This work extends the GMM-UBM system to use the dynamic information between feature vectors for writer identification by modeling the temporal sequencing among these underlying writing styles. This is performed by modeling writer identity with HMM with Gaussian observation model, which uses Gaussian distribution to model the distribution of feature vectors in each hidden handwriting style as GMM. This structure allows the HMM and GMM writer model to possess the same underlying writing style classes.

The specific training schematic for HMM writer model is showed in Fig. 2. First, the GMM writer model is trained with the GMM-UBM framework, which has been explained in Sect. 3.1. Second, the component parameters of the writer

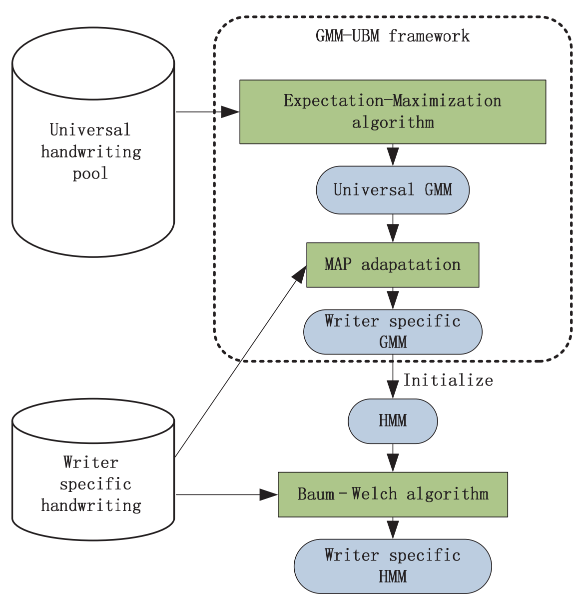
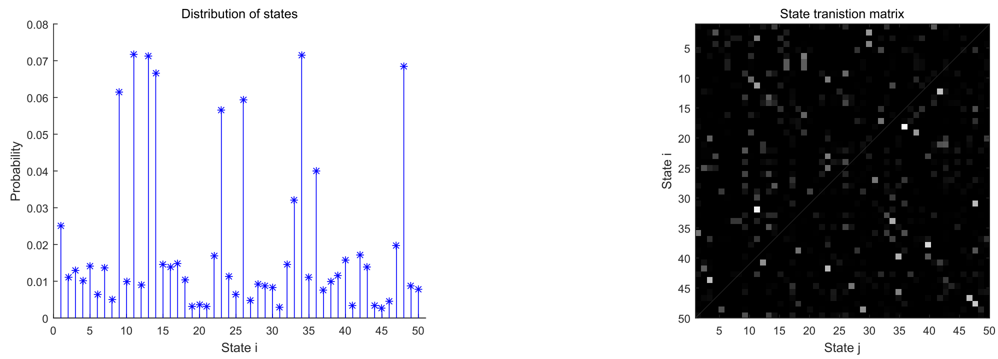


Fig. 2 The training procedure of the writer specific HMM.



(a) (b)

Fig. 3 Comparison of the statistics of states in GMM and HMM for one writer. (a) shows the distribution of states in the GMM-based system, (b) shows the probability of the state transitions in the HMM-based system.

specific GMM are used to set the observation parameters B of an HMM with Gaussian observation model. The number of hidden states in the HMM and the number of Gaussian component in GMM writer model are the same. Finally, the HMM are fitted into the writer specific handwriting using the Baum-Welch algorithm to acquire the writer specific HMM. During the iterative process, the parameters of emission density B are unchanged. Only the initial state distribution π and state transition probability matrix A are changed. This training method keeps the style classes learned with the GMM-UBM framework unchanged and learns the temporal structure between styles to use the dynamic information.

This training schematic restricts the underlying handwriting style classes in the writer specific GMM and HMM to be the same, which allows a clear investigation of the effect of the temporal modeling between styles. This is the motivation of using the HMM structure and the training method. The proposed system helps cope with the data sparsity problem for each writer. Setting the B parameters of HMM with the observation model parameters of GMM reduces the parameters of HMM to be fitted and less parameters need less data to fit.

According to the training method, the only difference between the proposed HMM-based writer model and GMM-based writer model trained with the GMM-UBM schematic is that different statistics of states are used as the prior information of the writer. Figure 3 shows the statistics of GMM and HMM for one writer in the experiments. In (a), the distribution of states in GMM is shown. We can see that the observations generated from which states are more preferred by this writer. While in (b), from the state transition matrix of HMM, we can see that the two time adjacent observations generated from which two states are more preferred by this writer.

During the testing phase, given a text $X = \{\mathbf{x}^k_{1:T_k}\}, k = 1:K$, where K is the number of strokes, the log-likelihood score for writer u is computed as

$$\log p(X|\lambda^{u,HMM}) = \sum_{k=1}^{K} \log p(\mathbf{x}^k_{1:T_k}|\lambda^{u,HMM}) \tag{6}$$

where $p(\mathbf{x}^k_{1:T_k}|\lambda^u)$ is computed with the forwards algorithm [11]. The writer giving the highest score is identified as the result. Commonly, the time between the the end point of one stroke and the start point of the next stroke is usually far longer than the time between two adjacent points in one stroke, so we assume the feature vectors in different strokes are independent.

4. Experiments

In the experiments, we use the IAM On-line English Handwritten Text Database(IAM-OnDB) [13]. IAM-OnDB consists of on-line data acquired from a white-board. The texts are of diverse nature, ranging from press and popular literature to scientific and religious writing. The database consists of more than 1700 handwritten forms from over 220 writers. It contains over 86,000 word instances with around 11,000 distinct words extracted from more than 13,000 text lines. For each writer, there are eight paragraphs of text. A paragraph of text consists of eight text lines in average. A text line contains 627 points and 24 strokes in average. The task in our experiments is to identify which person out of 200 individuals has written a given text.

4.1 Evaluation of the Time Information

The target of this paper is to further improve the performance of GMM-based system by incorporating the time information. In this section, the HMM trained with the proposed schematic is compared to the GMM-based systems. In order to investigate the effects of available data, the experiments are performed at text line level and paragraph level. In all the experiments, we use 4 paragraphs from each author's data for training, 2 paragraphs for validating and the rest 2 paragraphs for testing. Four fold cross validation is performed to get a more reliable result. The number of writers to be identified is 200. Both the number of states of HMM and the number of components of GMM are 50, which is the best performed parameter in [1]. During the MAP adaptation to acquire the writer specific GMMs, the

Table 1 Paragraph-level recognition result

Num	System	valid set average(%)	valid set std(%)	test set average(%)	test set std(%)
1	HMM-M0-V0	65.75	2.14	63.50	2.53
2	HMM-M1-V1	95.31	3.32	95.31	1.62
3	HMM-M2-V1	95.44	2.68	95.69	2.25
4	GMM-M1-V1-W2	89.44	2.91	90.69	2.83
5	GMM-M2-V1-W1	88.69	3.25	90.25	3.68
6	GMM-M2-V1-W2	93.75	2.96	94.50	3.14
7	GMM-M2-V1-W1[2]	88.19	4.75	88.56	3.90
8	GMM-M2-V1-W1[1]			85.00	
9	LDA+SVM [4]			93.39	

Table 2 Line-level recognition result

Num	System	valid set average(%)	valid set std(%)	test set average(%)	test set std(%)
1	HMM-M0-V0	48.81	3.30	48.00	2.84
2	HMM-M1-V1	73.97	4.08	73.84	2.57
3	HMM-M2-V1	77.17	4.55	76.28	2.56
4	GMM-M1-V1-W2	56.99	2.65	56.33	2.16
5	GMM-M2-V1-W1	60.79	3.78	60.90	3.53
6	GMM-M2-V1-W2	71.51	3.84	71.22	3.53
7	GMM-M2-V1-W1[2]	49.24	2.27	48.67	3.88

fixed relevance factor is set to 20. The right rate is used to indicate the performance, which is the percent of right recognized test samples.

Table 1 shows the paragraph-level writer identification results. In Table 2, the line-level results are displayed. We investigate 6 different systems in our experiments, i.e. 3 HMM-based systems and 3 GMM-based systems(numbered 1–6 in the two tables). For the naming of these methods, M and V denote the means and variances of the observation model for GMM or HMM and W denotes the weights of GMM. 0 denotes the parameters trained with the writer specific data directly, 1 denotes using the parameters of the UBM and 2 denotes using the MAP parameters. For example, HMM-M0-V0 represents the writer specific HMM is trained with the writer's data directly while HMM-M2-V1 denotes the means of the observation model are MAP means and the variances are the UBM values. In the paragraph-level experiments, our results are compared with the results in [1], [2], [4]. The line-level results are compared to the results in [2].

Firstly, we compare the performance of the GMM-M1-V1-W2 model and the HMM-M1-V1 model, which have the same observation model. Apparently, the HMM model performs better in both the paragraph-level and the line-level experiments. So modeling the hidden state sequence with Markov model to incorporate time information improves the performance. This viewpoint is further verified by the comparison of GMM-M2-V1-W1, GMM-M2-V1-W2 and HMM-M2-V1. These models possess the same observation model. Similar with the above results, the HMM performs better in the both level experiments. One interesting phenomenon in the paragraph-level experiment is that the accuracy difference between GMM-M2-V1-W2 and HMM-M2-V1 is about 1%, but the accuracy difference between GMM-M1-V1-W2 and HMM-M1-V1 is about 5%, which is much larger. It is common that higher the accuracy, more difficult the accuracy improvement. GMM-M1-V1-W2 has acquired the accuracy of 94.5%, which may be approaching the best accuracy allowed by the point-based features used in this work. So further improving the accuracy is more difficult than improving the accuracy from 90.69%. In the line level experiments, the accuracy difference between GMM-M2-V1-W2 and HMM-M2-V1 is clear. Comparing the recognition rates of HMM-M0-V0 and HMM-M1-V1, HMM-M2-V1, it is clear that the proposed training method has acquired significantly better performance than the HMM trained directly from the writer specific data in both the line level and the paragraph level experiments.

4.2 Evaluation of Model Order

The robustness of the proposed method under different model order is studied in this section. In this series of experiments no cross validation is performed to reduce the computational complexity. And only the recognition results on the test set are reported. The training set consists of four paragraphs, and the test set consists of two paragraphs. The experiments are performed on the text line level and paragraph level. The number of states N varies from 8 to 64. The studied systems are GMM-M2-V1-W2 and HMM-M2-V1, which are the best performed GMM and HMM system in the previous section. We also investigate the performance of the UA-UBM and US-HMM system in [9], in which the HMM with GMM emission probability are used. The UA-UBM system also only adapts the means as the original paper.

Table 3 shows the paragraph level recognizing precision while Table 4 illustrates the line level results. N denotes the number of states and M denotes the number of components in the GMM emission probability for the UA-UBM and US-HMM system. Comparing GMM-M2-V1-W2 and HMM-M2-V1, in the line level experiments, their accuracies are similar. In the line level experiments, at different number of states, HMM-M2-V1 performs significantly better than GMM-M2-V1-W2. Considering the results at both levels, HMM-M2-V1 performs better than GMM-M2-V1-W2. This shows that incorporating the time information using the proposed method improves the writer identification accuracy. In the case of the UA-UBM and US-HMM system, with the increase of the number of Gaussian distribution in each state and number of states, the US-HMM system outperforms the UA-UBM system in both level experiments. This may be attributed to that the UA-UBM system restricts the temporal sequence parameters of different HMM writer models to be same. Only adapting the means may not sufficient to model the dynamic difference between different writers' handwriting. Comparing UA-UBM, US-HMM and HMM-M2-V1, in both the line and paragraph level experiments and at different N, HMM-M2-V1 outperforms the two systems. This shows that the proposed training method is effective for training writer specific HMM with single Gaussian emission probability. The performance of the proposed HMM system is robust to selecting of model order and the amount of available test data.

Considering the complexities of the algorithms, for the training of GMM, each EM iteration complexity is $O(NTD)$, where N is the number of components, T is the length of the observation sequence, D is the samples' dimension. As indicated in [9], each Baum-Welch iteration complexity is $O(N^2TMD)$. For the proposed training method for HMM, first, tens of EM iterations are needed to train to the GMM UBM. Then one EM iteration is needed to acquire the writer specific GMM. Finally, tens of BW iterations are needed to acquire the writer specific HMM. However, only the π and A parameters are adapted during the BW iterations, so the values of $p(\mathbf{x}_t|z_t = i), t = 1 : T, i = 1 : N$ are unchanged and computed once in the first BW iteration. Other BW iterations' complexity is $O(N^2T)$. Apparently, additional BW iterations cause that the proposed method takes more time for training writer specific models than GMM-UBM system. In the case of UA-UBM system, tens of full BW iterations are needed to train the HMM UBM, which takes far more time than only adapting the π and A parameters. So with the same number of states, the training complexity of UA-UBM system is higher than the proposed system. If considering HMM with Gaussian emission probability, the test complexity of the proposed method and UA-UBM are the same. In similar with the training case, the test complexity of GMM-UBM is far less than the HMM based systems.

Table 3 Paragraph level results at different model order.

Method	M	N			
		8	16	32	64
HMM-M2-V1	1	89.25	93.75	95	94.25
GMM-M2-V1-W2		91.75	93.75	94	94.5
UA-UBM	1	86	87.25	90	88.75
	4	86	86	84	83.25
	8	84	80.75	80	79.5
US-HMM	1	45	73.5	88.5	91.75
	4	70	73	88	89.25
	8	78	82.5	89.5	85

Table 4 Line level results at different model order.

Method	M	N			
		8	16	32	64
HMM-M2-V1	1	63.69	73.27	75.64	76.4
GMM-M2-V1-W2		58.52	62.9	65.2	66.86
UA-UBM	1	52.71	56.52	60.29	58.67
	4	51.66	55.43	55.20	53.7
	8	52.68	54.52	53.47	52.91
US-HMM	1	32.13	50.23	62.59	68.25
	4	47.4	54.41	66.21	67.83
	8	53.62	59.65	68.02	63.71

4.3 Evaluation of Reducing Training Data

The third set of experiments measures the influence of using less data to train the HMMs. In this experimental setup, we reduce the amount of data available for training from four paragraphs to one paragraph in steps of one. The method HMM-M2-V1 in investigated. The meta parameters considered in this setup are the number of states (varied from 8 to 64) . In this series of experiments no cross validation is performed to reduce the computational complexity. And only the results on the test data set are reported. The test set each consists of two paragraphs. The experiments are performed on the text line level and the paragraph level. The rest of the experimental setup is identical to the ones of the previous two sets of experiments.

Table 5 shows the paragraph level results while Table 6 shows the line level results. N denotes the number of states for writer specific HMM. In the case of the paragraph level experiments, from training data of one paragraph to four paragraphs, the best performance acquired

Table 5 Paragraph level results at different amount of training data.

Amount of training data	N			
	8	16	32	64
One paragraph	83.5	86.5	69.25	49.5
Two paragraphs	89.75	91.5	90	90
Three paragraphs	92	93.5	94.5	94.25
Four paragraphs	90	93.25	95.5	94.25

Table 6 Line level results at different amount of training data.

Amount of training data	N			
	8	16	32	64
One paragraph	59.31	60.94	50.83	41.06
Two paragraphs	61.24	66.63	70.02	67.8
Three paragraphs	66.25	71.64	72.93	74.77
Four paragraphs	64.13	72.91	75.27	76.71

are 86.5%(N=16), 90%(N=32 or 64), 94.5%(N=32) and 95.5%(N=32). And when there are more than two paragraphs, the accuracy at $N = 32$ and $N = 64$ are very similar. The best performance of line level experiments are 60.94%(N=16), 70.02%(N=32), 74.77%(N=64) and 76.71%(N=64) with the increase of the training data. The results show that reducing the number of paragraphs for training from four to two paragraphs does not significantly reduce the writer identification rate. Analysing the influence of the model order, with the increase of the available training data, higher order are needed to acquire the best performance. When one paragraph of training data is available, $N > 16$ will cause dramatic decrease of identification performance, which indicates a result of data overfitting. With more available training data, too low the model order will not be able to model the information contained in the data sufficiently.

5. Conclusion

In this paper, we have analysed the influence of dynamic information for text-independent online writer identification by modeling each writer's handwriting with an HMM. Based on the well performed GMM-UBM system, a new training schematic is proposed for writer specific HMMs in which the observation parameters of the HMM are set with the parameters of the GMM and only the left parameters are fitted. So the dynamic information is incorporated by modeling the distribution of state transitions. This helps cope with the data sparsity problem of each writer's handwriting.

In the experiments, the proposed HMM-based system performs better than the popular GMM-based system in both the paragraph and line level experiments, which indicates that the time information is also important for text-independent tasks. The performance of the UA-UBM and US-HMM system [9] are also investigated. The results show that the proposed system performs better. This indicates the observation model parameters of GMM trained with the GMM-UBM framework are good initial estimates of the B parameters of single Gaussian HMM and the proposed training method in an effective method for training writer specific HMMs. Analysing the training and texting complexity of the GMM-UBM, UA-UBM and the proposed method, if with the same number of states, the training and testing complexity of GMM-UBM is far less than the other two systems. However, the training complexity of the proposed method is far less than UA-UBM system.

References

[1] M. Liwicki, A. Schlapbach, H. Bunke, S. Bengio, J. Mariéthoz, and J. Richiardi, "Writer identification for smart meeting room systems," In Proc. 7th IAPR Workshop on Document Analysis Systems, volume 3872 of LNCS, p.186, 2006.

[2] A. Schlapbach, M. Liwicki, and H. Bunke, "A writer identification system for on-line whiteboard data," Pattern Recognit., vol.41, no.7, p.2381–2397, 2008.

[3] M. Bulacu and L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," IEEE Trans. Pattern Anal. Mach. Intell., vol.29, no.4, pp.701–717, 2007.

[4] A. Shivram, C. Ramaiah, and V. Govindaraju, "A hierarchical bayesian approach to online writer identification," Iet Biometrics, vol.2, no.4, pp.191–198, 2013.

[5] M. Bulacu, L. Schomaker, and L. Vuurpijl, "Writer identification using edge-based directional features," 2003 Seventh International Conference on Document Analysis and Recognition, p.937, 2003.

[6] J. Fierrez, J. Ortega-Garcia, D. Ramos, and J. Gonzalez-Rodriguez, "Hmm-based on-line signature verification: feature extraction and signature modeling," Pattern Recognit. Lett., vol.28, pp.2325–2334, 2007.

[7] M. Martinez-Diaz, J. Fierrez, M. Freire, and J. Ortega-Garcia, "On the effects of sampling rate and interpolation in hmm-based dynamic signature verification," Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), pp.1113–1117, 2007.

[8] D. Muramatsu and T. Matsumoto, "An hmm on-line signature verifier incorporating signature trajectories," International Conference on Document Analysis and Recognition, pp.438–438, 2003.

[9] E.A. Rua and J.L.A. Castro, "Online signature verification based on generative models.," IEEE Trans. Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society, vol.42, no.4, pp.1231–42, 2012.

[10] D.A. Reynolds, T.F. Quatieri, and R.B. Dunn, "Speaker verification using adapted gaussian mixture models," Digital Signal Processing, p.19–41, 2000.

[11] L.R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," Proc. IEEE, vol.77, no.2, pp.257–286, 1989.

[12] K.P. Murphy, Machine learning : a probabilistic perspective, MIT Press, 2012.

[13] M. Liwicki and H. Bunke, "Iam-ondb - an on-line english sentence database acquired from handwritten text on a whiteboard," 2005 Eighth International Conference on Document Analysis and Recognition, pp.956–961, vol.2, 2005.

Yabei Wu received the B.S. and M.S. degrees in Information and Electronic Engineering from Xidian University and National University of Defense Technology in 2010 and 2013, respectively. He is now a Ph.D. candidate on Science and Technology on Automatic Target Recognition Laboratory, National University of Defense Technology.

Huanzhang Lu received the M.S. and Ph.D. degrees from National University of Defense Technology in 1990 and 1994, respectively. He is now a professor at National University of Defense Technology.

Zhiyong Zhang received the M.S. and Ph.D. degrees from National University of Defense Technology in 2001 and 2006, respectively. He is now a vive professor at National University of Defense Technology.