Writer Adaptive Training and Writing Variant Model Refinement for Offline Arabic Handwriting Recognition

Philippe Dreuw, David Rybach, Christian Gollan, and Hermann Ney
RWTH Aachen University
Human Language Technology and Pattern Recognition
Ahornstr 55, D-52056 Aachen, Germany
<surname>@cs.rwth-aachen.de

Abstract

We present a writer adaptive training and writer clustering approach for an HMM based Arabic handwriting recognition system to handle different handwriting styles and their variations. Additionally, a writing variant model refinement for specific writing variants is proposed.

Current approaches try to compensate the impact of different writing styles during preprocessing and normalization steps.

Writer adaptive training with a CMLLR based feature adaptation is used to train writer dependent models. An unsupervised writer clustering with Bayesian information criterion based stopping condition for a CMLLR based feature adaptation during a two-pass decoding process is used to cluster different handwriting styles of unknown test writers.

The proposed methods are evaluated on the IFN/ENIT Arabic handwriting database.

1. Introduction

In this paper, we describe our writer adaptive training and multi-pass decoding system for off-line Arabic handwriting, and present systematic results on the IFN/ENIT database [10].

Most state-of-the-art HMM based handwriting recognition systems are single-pass training and decoding systems [8], some multi-pass handwriting recognition systems have been recently presented in [2, 5]. Opposed to the writer specific system presented in [7], where a system previously trained on a large general off-line handwriting database is adapted by writer specific data of handwritten manuscripts from the 20th century, our training systems consists of a writer adaptive training process, and our decoding system consists of two subsystems each using a differently trained character model.

Due to ligatures and diacritics in Arabic handwriting, the same Arabic word can be written in several writing variants, depending on the writer's handwriting style. Similar to dictionary learning in automatic speech recognition (ASR) [11], where a-priori knowledge about specific pronunciation variants can be used for acoustic model refinement, the a-priori probability of observing a specific writing variant can be used in handwriting recognition for writing variant model refinement during training and decoding. Additionally, during training, the writing variants can be used in a supervised manner, which would correspond to a phoneme transcribed corpora in ASR.

A character based clustering of writing styles with a self-organizing map is presented in [13]. Unsupervised clustering that estimates Gaussian mixture models for writing styles in combination with a maximum likelihood linear regression (MLLR) based adaptation of the models is presented in [5, 12]. In [1], a writer identification and verification approach using local features is presented.

Our system uses a writer adaptive training method using constrained maximum likelihood linear regression (CMLLR) based writer dependent adaptation of the features instead of the models to train writer specific models. During recognition, in a first pass, we estimate in an unsupervised writer clustering step with Bayesian information criterion based stopping condition [6] clusters for the unknown writers and their writing styles. In the second pass, we use these clusters for a writer dependent estimation of the CMLLR based feature adaptation. These steps are described in the following sections.

2. System Overview

We are searching for an unknown word sequence $w_1^N := w_1, \ldots, w_N$, for which the sequence of features $x_1^T := x_1, \ldots, x_T$ best fits to the trained models. We maximize

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.9

the posterior probability $p(w_1^N|x_1^T)$ over all possible word sequences w_1^N with unknown number of words N. This is modeled by Bayes' decision rule:

$$\hat{w}_1^N = \arg\max_{w_1^N}\{p^\gamma(w_1^N)p(x_1^T|w_1^N)\} \quad (1)$$

with γ a scaling exponent of the language model.

Here we propose a writing variant model refinement of our character model in Equation 2:

$$p(x_1^T|w_1^N) \approx \max_{v_1^N|w_1^N}\{p^\alpha_{\theta_{\text{pm}}}(v_1^N|w_1^N)p^\beta_{\theta_{\text{em,tp}}}(x_1^T|v_1^N,w_1^N)\} \quad (2)$$

with v_1^N a sequence of unknown writing variants, α a scaling exponent of the writing variant probability depending on a parameter set θ_{pm}, and β a scaling exponent of the character model depending on a parameter set $\theta_{\text{em,tp}}$ for emission and transition model.

2.1. Feature Extraction

Without any preprocessing of the input images, we extract simple appearance-based image slice features X_t at every time step $t = 1, \cdots, T$ which are augmented by their spatial derivatives in horizontal direction $\Delta = X_t - X_{t-1}$. In order to incorporate temporal and spatial context into the features, we concatenate 7 consecutive features in a sliding window, which are later reduced by a PCA transformation matrix to a feature vector x_t.

2.2. Writing Variant Model Refinement

Due to ligatures and diacritics in Arabic handwriting, the same Arabic word can be written in several writing variants, depending on the writer's handwriting style.

During training, a corpus and lexicon with supervised writing variants instead of the commonly used unsupervised writing variants can be used in Viterbi training. Obviously, the supervised writing variants in training can lead to better trained character models only if the training corpora have a high annotation quality.

During the decoding steps, the writing variants can only be used in an unsupervised manner. Usually, the probability $p(v|w)$ for a variant v of a word w is considered as equally distributed [4]. Here we use the count statistics as probability

$$p(v|w) = \frac{N(v,w)}{N(w)} \quad (3)$$

where the writing variant counts $N(v,w)$ and the word counts $N(w)$ are estimated from the corresponding training corpora, and represent how often these events were observed. Note that $\sum_{v'} \frac{N(v',w)}{N(w)} = 1$. Additionally, the scaling exponent α of the writing variant probability of Equation 2 can be adapted in the same way as it is done for the language model scale γ in Equation 1.

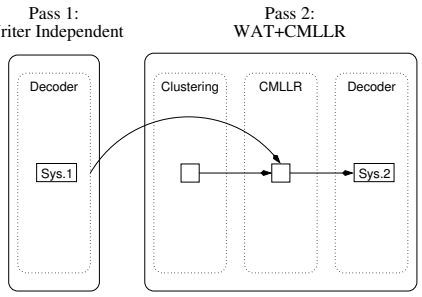

Figure 1. Illustration of the two-pass decoding process.

2.3. Model Length Estimation

After a first pass training, the number of states per character can be estimated for a given training alignment. Using the model length estimation (MLE) method as proposed in [4], the number of states S_c for each character c is updated by

$$S_c = \frac{N_{x,c}}{N_c} \cdot f_P \quad (4)$$

with S_c the estimated number states for character c, $N_{x,c}$ the number of observations aligned to character c, N_c the character count of c seen in training, and f_P a character length scaling factor.

2.4. Writer Adaptive Training

Our hidden Markov model (HMM) based handwriting recognition system is Viterbi trained and uses a lexicon with multiple writing variants, where the white-spaces between the pieces of Arabic words are explicitly modeled as proposed in [4].

Writer variations are compensated by writer adaptive training (WAT) using constrained maximum likelihood linear regression (CMLLR) [6]. The available writer labels of the IFN/ENIT database are used in training to estimate the writer dependent CMLLR feature transformations. The parameters of the writer adapted Gaussian mixtures are trained using the CMLLR transformed features. It can be seen from the writer statistics in Table 1 that the number of different writers in set *e* is higher than in all other folds, and thus the variation of handwriting styles.

3. Decoding Architecture

First Pass. The recognition is performed in two passes, as depicted in Figure 1. System 1 performs the initial and independent recognition pass. The automatic transcriptions are required for the text dependent writer adaptation in the next step.

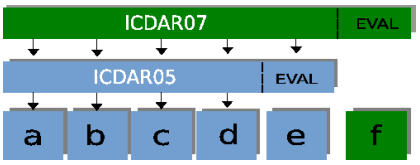

Figure 2. IFN/ENIT corpora splits used in 2005 and 2007.

Table 1. IFN/ENIT corpus statistics.

set	number of writers	number of samples
a	102	6537
b	102	6710
c	103	6477
d	104	6735
e	505	6033
Total	916	32492

Second Pass - Writer Adaptation. The decoding in the second pass is carried out using CMLLR transformed features. The segments to be recognized are first clustered using a generalized likelihood ratio clustering with Bayesian Information Criterion (BIC) based stopping condition [3]. The segment clusters act as writer labels required by the unsupervised adaptation techniques. The CMLLR matrices are calculated in pass two for every estimated writer cluster and are used for a writer dependent recognition in System 2, which uses the models from the writer adaptive training of subsection 2.4.

It should be noted that all experiments in the following section were done without any pruning, and thus the improvement of the system accuracy is due to the proposed refinement methods only.

4. Experimental Results

The experiments are conducted on the IFN/ENIT database [10]. The database is divided into four training folds with an additional fold for testing [9]. The current database version (v2.0p1e) contains a total of 32492 Arabic words handwritten by 916 writers, and has a vocabulary size of 937 Tunisian town names. Additionally, the submitted systems to the ICDAR 2007 competition [8] were trained on all datasets of the IFN/ENIT database and evaluated for known datasets. Here, we follow the same evaluation protocol as in ICDAR 2005 and 2007 competition (see Figure 2).

4.1. Writing Variant Model Refinement

In Table 2 we analyze the impact of supervised writing variants (SWV) in training. The word-error-rate (WER) and the character-error-rate (CER) is decreased using a corpus and training lexicon with supervised writing variants.

Table 2. Comparison of supervised and unsupervised writing variants in training.

Train	Test	unsupervised		supervised	
		WER[%]	CER[%]	WER[%]	CER[%]
abc	d	11.60	3.88	11.00	3.66
abd	c	12.95	4.60	11.41	3.97
acd	b	11.98	3.91	11.16	3.65
bcd	a	12.33	4.26	11.93	4.27
abcd	e	24.60	9.34	22.58	8.39
abcde	e	11.74	4.37	11.37	4.17

During decoding, we can observe in Figure 3 that the unsupervised writing variant scaling has hardly any influence on the system performance in the cross validation setups, whereas the error rate can be decreased by 3% relative on the evaluation set. Note that the experiment setup using the training sets *abcd* and test set *e* has the highest number of writing variants in the training sets, and also the highest number of different writers in the test set (see Table 1). Here, a writing variant scaling of $\alpha = 15.0$ reduces the impact of rare writing variants, and increases the impact of frequent writing variants during the decoding search. In total, the error rate can be decreased by 11% relative from 24.60% to 21.86% with the proposed writing variant model refinements.

4.2. Model Length Estimation

The alignments of the supervised writing variants (SWV) trained models are used to estimate the number of states per character. The necessity of this character dependent model length estimation is visualized in Figure 4, where we use R-G-B background colors for the 0-1-2 HMM states, respectively, from right-to-left: the bottom row images visualize an alignment of our SWV trained baseline system (left) in comparison to the proposed MLE system (right).

By estimating character dependent model lengths, the overall mean of character length changed from 7.89px (i.e. 2.66 px/state) to 6.18px (i.e. 2.06px/state) when downscaling the images to 16px height while keeping their aspect-ratio. Thus every state of a MLE character model has to cover less pixels due to the relative reduction of approx. 20% pixels.

After estimating the number of states per character, we retrained the SWV system using the MLE adapted training lexicon. The results in Table 3 show that the SWV trained

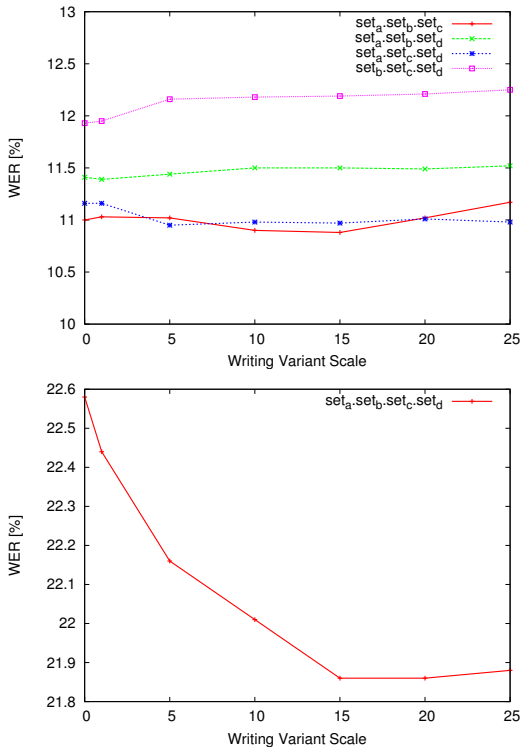

Figure 3. Empirical optimization of the writing variant scale α on the cross folds and verification on the development set.

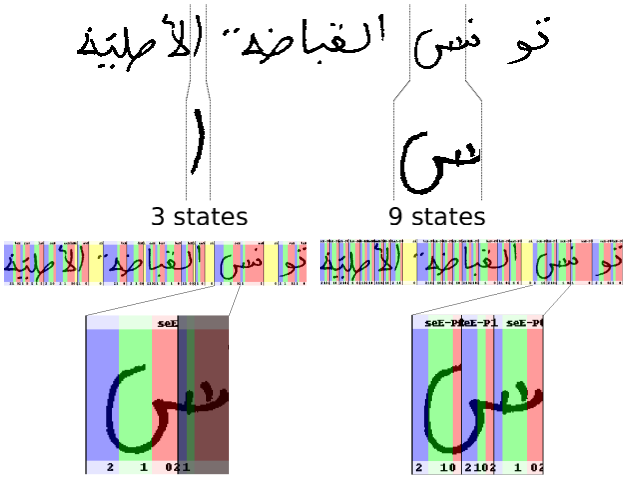

Figure 4. Top: more complex characters should be represented by more states. Bottom: after the MLE, frames previously aligned to a wrong neighboring character model (left, black shaded) are aligned to the correct character model (right).

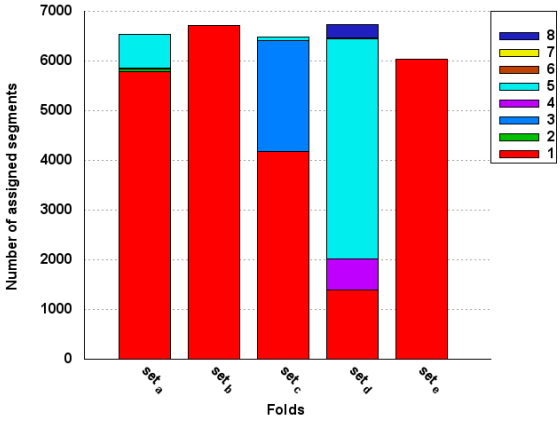

Figure 5. Histograms for unsupervised clustering over the different test folds and their resulting unbalanced segment assignments.

model with additional MLE decreases the error rate in all testing conditions, and even outperforms the system presented in [4].

4.3. Writer Adaptation

First Pass - Writer Adaptive Training. Using the supervised writing variants lexicon, the writer adaptive trained (WAT) models (c.f. subsection 2.4) can also be used as a first pass decoding system. The results in Table 3 show that the system performance cannot be improved without any writer clustering and adaptation of the features during the decoding step.

Second Pass - CMLLR based Writer Adaptation. The decoding in the second pass is carried out using the CMLLR transformed features.

To show the advantage of using CMLLR based writer adapted features in combination with WAT models, we estimate in a first *supervised* experiment the CMLLR matrices directly from the available *writer labels* of the test folds. The matrices are calculated for all writers in pass two and are used for a writer dependent recognition in System 2, which uses the WAT models from subsection 2.4. Note that the decoding itself is still unsupervised!

In the unsupervised adaptation case, the unknown writer labels of the segments to be recognized have to be estimated first using BIC clustering. Again, the CMLLR matrices are calculated in pass two for every estimated cluster label and are used for a writer dependent recognition in System 2, which uses the WAT models from subsection 2.4.

Table 3 shows that the system accuracy could be improved by up to 33% relative in the supervised-CMLLR adaptation case. In the case of unsupervised writer clustering, the system accuracy is improved in one fold only.

Table 3. Comparison of MLE, WAT, and CMLLR based feature adaptation using unsupervised and supervised writer clustering.

Train	Test	WER[%]				
		1st pass			2nd pass	
		SWV	+MLE	+WAT	WAT+CMLLR	
					unsup.	sup.
abc	d	10.88	**7.83**	7.54	7.72	**5.82**
abd	c	11.50	**8.83**	9.09	9.05	**5.96**
acd	b	10.97	**7.81**	7.94	7.99	**6.04**
bcd	a	12.19	**8.70**	8.87	8.81	**6.49**
abcd	e	21.86	**16.82**	17.49	17.12	**11.22**
abcde	e	11.14	**7.74**	8.37	7.79	**5.12**

If we look at the cluster histograms in Figure 5 it becomes clear that the unsupervised clustering is not adequate enough. Each node in our clustering process as described in [3] is modeled as a multivariate Gaussian distribution $\mathcal{N}(\mu_i, \Sigma_i)$, where μ_i can be estimated as the sample mean vector and Σ_i can be estimated as the sample covariance matrix. The estimated parameters are used within the criterion as distance measure, but more sophisticated features than the PCA reduced sliding window features seem necessary for a better clustering.

Opposed to the supervised estimation of 505 CMLLR transformation matrices for the evaluation setup with training sets *abcd* and set *e* (c.f. Table 1), the unsupervised writer clustering could estimate only two clusters being completely unbalanced, which is obviously not enough to represent the different writing styles of 505 writers. Due to the unbalanced clustering and only a small number of clusters, all other cases are similar to the usage of the WAT models only (c.f. Table 3).

However, the supervised-CMLLR adaptation results show that a good writer clustering can bring the segments of the same writer together and thus improve the performance of the writer adapted system.

5. Conclusions

We presented an HMM based system for off-line Arabic handwriting recognition which uses writer adaptive training and a two-pass decoding step with unsupervised writer clustering and CMLLR based feature adaptation. The advantages of the proposed methods were shown on the IFN/ENIT corpus.

The proposed writing variants model refinement in combination with a character dependent model length estimation could improve the system accuracy for all conditions.

The impact of different writing styles was handled by writer adaptive training in combination with unsupervised writer clustering and CMLLR based feature adaptation. The supervised writer adaptation demonstrated the potential of these techniques, and the analysis of better writer clustering techniques, more sophisticated features, and distance measures to be used within the clustering will be interesting for future work. In particular, and to the best of our knowledge, the presented results outperform all error rates reported in the literature.

Acknowledgements. This work was partly realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation.

References

[1] A. Bensefia, T. Paquet, and L. Heutte. Handwritten document analysis for automatic writer recognition. *Electronic Letters on CVIA*, 5(2):72–86, May 2005.

[2] R. Bertolami and H. Bunke. Hmm-based ensamble methods for offline handwritten text line recognition. *Pattern Recognition*, 41:3452–3460, 2008.

[3] S. S. Chen and P. S. Gopalakrishnan. Speaker, environment and channel change detection and clustering via the bayesian information criterion. pages 127–132, 1998.

[4] P. Dreuw, S. Jonas, and H. Ney. White-space models for offline arabic handwriting recognition. In *ICPR*, Tampa, Florida, USA, Dec. 2008.

[5] G. A. Fink and T. Plötz. Unsupervised estimation of writing style models for improved unconstrained off-line handwriting recognition. In *IWFHR*, La Baule, France, Oct. 2006.

[6] M. J. F. Gales. Maximum likelihood linear transformations for HMM-based speech recognition. 12(2):75 – 98, Apr. 1998.

[7] E. Indermühle, M. Liwicki, and H. Bunke. Recognition of handwritten historical documents: Hmm-adaptation vs. writer specific training. In *ICFHR*, pages 186–191, 2008.

[8] V. Märgner and H. E. Abed. ICDAR 2007 Arabic handwriting recognition competition. In *ICDAR*, volume 2, pages 1274–1278, Sept. 2007.

[9] V. Märgner, M. Pechwitz, and H. Abed. ICDAR 2005 Arabic handwriting recognition competition. In *ICDAR*, volume 1, pages 70–74, Seoul, Korea, Aug. 2005.

[10] M. Pechwitz, S. S. Maddouri, V. Mägner, N. Ellouze, and H. Amiri. IFN/ENIT-database of handwritten Arabic words. In *CIFED*, Hammamet, Tunis, Oct. 2002.

[11] T. Sloboda and A. Waibel. Dictionary learning for spontaneous speech recognition. In *ICSLP*, pages 2328–2331, Philadelphia, PA, USA, Oct. 1996.

[12] A. Vinciarelli, A. Vinciarelli, S. Bengio, and S. Bengio. Writer adaptation techniques in hmm based off-line cursive script recognition. *Pattern Recognition Letters*, 23:2002, 2002.

[13] V. Vuori. Clustering writing styles with a self-organizing map. In *IWFHR*, pages 345–350, 2002.