Published version: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7333807

Training an Arabic handwriting recognizer without a handwritten training data set

Irfan Ahmad [1, 2]
[1] Information and Computer Science Department
KFUPM, Dhahran Saudi Arabia
irfanics@kfupm.edu.sa

Gernot A. Fink [2]
[2] Department of Computer Science
TU Dortmund University, Germany
gernot.fink@tu-dortmund.de

***Abstract* — Handwritten text recognition is an active research area in pattern recognition. One of the prerequisites of setting up a handwritten text recognizer is to train them using, mostly, large amounts of labeled training data. In the current paper we report our work on handwritten text recognition using no handwritten training set. We investigate different approaches including, computer generated text in different typefaces as training data, unsupervised adaptation, and using recognition hypothesis on the test sets as training data. Results from handwritten Arabic word recognition task show that the approach is promising with good recognition rates.**

Keywords — Handwritten text recognition, hidden Markov models, training data, efficient training, HMM adaptation, OCR.

I. Introduction And Related Work

Handwritten text recognition is a challenging task. A variety of features and classifiers have been investigated for handwritten text recognition [1], [2]. A core step in setting up a text recognizer is to train the classifier. Availability of sufficient number of training samples for each class is important for adequate training. These classes can represent characters, character shapes, strokes or other suitable representations of text. To assist the research in the area, benchmark databases are developed to provide data for training and calibrating the recognizer and to evaluate a recognizer's performance on the evaluation sets [3]–[6]. In order to assure that adequate training can be done, huge amounts of data are collected for training and then, they need to be labeled correctly which is a laborious and time consuming task. Researchers in this area generally believe that the quantity and quality of training data is as important as developing effective features and classifiers [7], [8]. Some semi-supervised techniques to label the data have been investigated so as to alleviate the problem of manually labeling the data [9], but the problem is far from being solved in this respect.

Researchers have identified this issue and some work has been reported to deal with situations involving relatively small training sets. The most notable approach is to synthesize training data from an existing training set which is regarded not big enough to adequately train the recognizer [10]. Varga and Bunke [8] presented a work on expanding the training set by adding synthesized data in addition to the original handwritten data. They presented a perturbation model to synthesize text lines from handwritten samples by performing a number of geometrical distortions in addition to thinning and thickening of strokes. Experiments conducted for offline handwritten text recognition showed that adding the synthetic data to the original training set led to improvements in recognition rates. It was claimed that improvements in recognition rates were possible, both when the original training set is small as well as large although it is easier to improve in situations when the training set is small. Elarian et al. [11] presented two approaches to synthetically generate additional training data from a small set of handwritten Arabic text images. The results from a closed vocabulary handwritten Arabic text recognition task showed that addition of synthetic data to the original training set improved the text recognition results. Miyao and Maruyama [12] presented their work on using synthesized characters, in addition to the real samples, in order to improve the training of the Japanese Hiragana character recognition system. Characters were synthesized by applying an affine transformation to each stroke of the on-line characters. Interested readers can refer to [10] for further reading on the use of synthesized data for improving the text recognition performance.

Another notable approach of using few samples of labeled training data is based on semi-supervised training. Frinken et al. [13] used the concept of "co-training" where few labeled samples are used to initialize two different systems. The two systems were then iteratively trained by recognizing the unlabeled training data followed by filtering the good recognition results and providing it as training data for the other system in the next iteration. Recently, Kozielski et al. [14] presented work aimed at training the recognizer using only unlabeled training data. They fine tune the recognizer by iteratively generating hypothesis and then in-turn training the recognizer on the hypothesized data. The most challenging aspect was to initialize the recognizer and to generate an initial hypothesis of the unlabeled training data. To address this, they used the language models along with word-image length information to make an initial guess of the word in the first iteration. They also used some heuristics based approach to train white-space models separately from the character models. Experiments conducted on word recognition task on two separate datasets showed the effectiveness of the approach. The results, although lower than systems trained on labeled training data, were quite promising.

According to the best knowledge of the authors, there is no work reported in the literature to deal with situations where no handwritten training set is available. In this paper, the authors present their work on handwritten Arabic text recognition without the use of any handwritten training set. Several approaches were investigated on a closed vocabulary handwritten Arabic text recognition task. To initialize the recognition system at some reasonable level (which is a challenging step in such problems), we use computer generated machine printed text as training data and later perform unsupervised HMM adaptation during recognition. As Arabic script is cursive both in machine printed and handwritten forms, using computer generated machine printed text for training and

adapting it for handwritten text recognition task proves to be a promising approach. This direction of research can have favorable implications in the future by greatly minimizing, if not complete eliminating, the need of training set for handwritten text recognition tasks.

The rest of the paper is organized as follows: In Section II, we present the different approaches we investigated for text recognition without the need for a handwritten training set. In Section III, we present the experimental results and the discussions. Finally in Section IV, we present the conclusions from our work.

II. Approaches To Handwritten Arabic Text Recognition Without Handwritten Training Set

In this section, we present our ideas for handwritten Arabic text recognition in situations where no handwritten training set is available. We organize our discussions into approaches such that the next approach builds on the approach before it. Overall, we investigated the following approaches:

A. Training the classifier using computer generated text

The first approach we investigate is to use computer generated text as training data for the classifier. We generate text in a number of different font typefaces and train separate classifiers using generated text in each typeface. We were interested to know if a classifier trained on computer generated text would be good enough to recognize handwritten text or not. As Arabic script is cursive, both in the machine printed and the handwritten form, the machine printed text does have some resemblance to the handwritten text although with a smaller degree of variability. This aspect might be useful to initialize the handwritten text recognizer using machine printed text as training data. We were also interested to know if the classifiers trained on computer generated text perform similar irrespective of the typeface of the text or do visually complex-looking typefaces work better than simple-looking typefaces or vice-versa.

B. Training the classifier using computer generated text in multiple typefaces

Using this setup, we were interested to investigate how good a recognizer can perform when it is trained on computer generated text on multiple typefaces instead of training a classifier on only one typeface. Does a classifier trained on text from multiple typefaces perform better or worse than the classifiers trained on text from a single typeface? There is a good reason to believe that the recognizer may perform better when trained on multiple typefaces as this may, to some extent, enable us to model the variability of handwriting better as compared to the recognizer trained on text from only one typeface.

C. Using unsupervised HMM adaptation during recognition

Unsupervised HMM adaptation techniques recalibrate the trained parameters based on the new data they see during recognition. HMM adaptation techniques have successfully been used to improve the performance of speech recognition systems [15], [16]. It is used mainly for speaker adaptation. Instead of training the recognizer for a particular speaker which may need a lot of data from a particular speaker, a small amount of speaker specific data can be used to adapt the model parameters of a general recognizer to fit the speaker specific characteristics. The same idea of HMM adaptation has been successfully extended to the domain of text recognition. It has been used for adaptation of handwritten text recognizer for new writers [17]. HMM adaptations techniques were applied for adapting a multi-font text recognizer to a specific font text recognition task [18]. In the present work, we investigated the use of HMM adaptation to adapt a recognizer trained on printed text to handwritten text recognition task. As we do not use any labeled training data, we perform unsupervised HMM adaption during recognition.

In general, model parameters related to the data part are adapted (mixture means μ and variances Σ) and the model length and the transition probabilities are not modified. The task of adaptation is to find the new model parameters $\hat{\theta}$ by fine tuning the original model parameters θ to maximize the likelihood of adaptation data O:

$$\hat{\theta} = \arg\max_{\theta} p(\theta|O)$$

Maximum Likelihood Linear Regression (MLLR) is one of the most common techniques used for HMM adaptation. In order to adjust the mixture means and variances to better fit the adaptation data, MLLR estimates linear transformations for them. Transformations are tied across several Gaussians in order to robustly estimate the transformations, given the availability of little adaptation data. A group of Gaussians that share the same transform are termed as a regression class. For more details on HMM adaptations using MLLR, readers can refer to [15], [16].

D. Using recognition hypothesis on test set as training data

In this approach, we generate recognition hypothesis for the handwritten text of the test set using the system developed by the previous approaches. Next, we use this recognition hypothesis to train the classifier. Once the handwritten text has been labeled "reasonably" well, using this test data in-turn for training can prove to be an effective approach and may perform better than the previous approaches. In this case, the previous approaches can be regarded as initialization steps to start-up the recognizer. Clearly for this approach to work it will be important that the recognition hypothesis generated at this stage is somewhat reliable. Training on poorly hypothesized data can in fact do more harm than good and can even perform worse than the previous approaches as the classifier will get trained on a huge amount of wrongly labeled data [7]. Thus we need to limit, if not completely remove, the mislabeled data from the correctly hypothesized data. To address this issue, we remove the bottom five percent of the hypothesized data based on the length-normalized scores. Another approach which we have not yet investigated is to use the recognition hypothesis on the test set together with the computer generated text as the training data. This will require a careful mix of the two sets so as not to over-train a classifier on one type of data.

E. Improving the recognition using multiple iterations of the previous step

If the previous step works reasonably well, then we can improve the recognizer's performance by iteratively feeding the improved recognition hypothesis as training data (after removing the images having the worst length-normalized recognition scores) which in-turn can lead to better recognition. After certain iterations, the recognition performance may reach an improvement threshold and further iterations may not necessarily improve much.

III. Experiments and Results

In this section we present the experiments, the results, and the discussions on setting up a recognizer for an Arabic handwritten text recognition task when no handwritten training

set is used. We will present our experiments in steps following the same order as the discussions in Section II. We first present the experiments on using recognizers trained on computer generated text on single typefaces. This is followed by experiments using a recognizer trained on computer generated text on multiple typefaces. Next, we present the text recognition using unsupervised HMM adaptation. Finally, we present experiments related to the use of recognition hypothesis on handwritten text images of the test set as training data.

A. Recognition task

Our task is offline Arabic handwritten word recognition using the IFN/ENIT database [5]. The database consists of handwritten word images of Tunisian cities and towns divided into seven sets *a – f*, and *s*. As we are not training our recognizer on handwritten data, we do not use sets *a – c* which are commonly used for training. We only use sets *d – f*, and *s* individually for evaluation. The lexicon size is 937 names, but some names have two or more variations (mainly due to *ligature* models and optional *shadda* diacritics).

B. Sub-Character HMM recognizer for text recognition

For the experimentation we use the Arabic sub-character model based HMM recognizer as presented in [19], [20]. We use sub-character model based HMM system as it seems to be more robust and effective especially under constrained training environments. Our recognizer is a continuous HMM system built using HTK tools [21]. It uses 97 sub-character HMMs to model all the characters and their shape variations. Only uniform initialization (sometimes also termed as flat-start) on the training set was performed. We extracted nine geometrical features (the average number of ink pixels, the number of black-white transitions, the distance of the upper contour, the lower contour, and the center-of-gravity of the ink-pixels from the writing line, the orientation of the upper contour, the lower contour, and the center-of-gravity of the ink pixels) from each frame sliding over the word images. These features are adapted from [22] and [23]. In addition, we appended nine derivative features to the extracted features such that the dimension of the feature vector is 18.

C. Experimental details and discussion

In the first set of experiments, we use computer generated text from single typefaces to train our recognizer. To generate text, we use the IFN/ENIT lexicon with all its variations. Using the IFN/ENIT lexicon was not a prerequisite, as the only thing we needed was to have some training samples to train the different HMM models. Since our recognition task was on the IFN/ENIT database, we generated text using its lexicon. We generated 1929 images for each typeface corresponding to 1929 entries in the dictionary i.e. we generated one sample per entry for eight different typefaces. Samples of computer generated text in different typefaces along with handwritten text images from the IFN/ENIT database for the same word are shown in Fig. 1. We trained eight different recognizers, each trained on text from one typeface. Once the recognizer was trained, we evaluated the recognizer by recognizing word images from set *d* of the IFN/ENIT database. The evaluation results are shown in Table I. The results are shown in terms of *Word Recognition Rate* (*WRR*).

Computer generated Images *(typeface)*	**Samples from IFN/ENIT database**
القلعة الكبرى *(Arabic Typesetting)*	
القلعة الكبرى *(Diwani)*	
القلعة الكبرى *(Naskh)*	
القلعة الكبرى *(Rekaa)*	
القلعة الكبرى *(Tahoma)*	
القلعة الكبرى *(Thuluth)*	
القلعة الكبرى *(Traditional Arabic)*	
القلعة الكبرى *(Zarnew)*	
الطويرف *(Arabic Typesetting)*	
الطويرف *(Diwani)*	
الطويرف *(Naskh)*	
الطويرف *(Rekaa)*	
الطويرف *(Tahoma)*	
الطويرف *(Thuluth)*	
الطويرف *(Traditional Arabic)*	
الطويرف *(Zarnew)*	

Fig. 1. Samples of computer generated images in different typefaces along with samples from handwritten text images from the IFN/ENIT database for two different city names.

From the results shown in the table we have following observations: Although the results were not entirely disappointing, in general the recognition rates were very low for most of the typefaces, which is understandable. The character glyphs for computer generated texts are very regular with only one fixed pattern. It is very difficult for the recognizer to train the models which can cope with the huge variations found in human handwriting. Nevertheless, some typefaces did relatively well; the recognizer trained on the *Naskh* typeface was able to achieve 26.92% WRR, i.e. it was successful in recognizing one-fourth of the total word images from set *d*. Another interesting observation was that, although the recognizer trained on the simple typefaces like *Tahoma* did worst, the recognizers trained on very complex typefaces like *Rekaa* and *Diwani* did poorly as well.

TABLE I: WRR ON SET 'D' OF IFNENIT DATABASE FOR RECOGNIZER TRAINED ON SINGLE TYPEFACE.

Typeface	**WRR (%)**
Arabic Typesetting	11.25

Diwani	10.01
Naskh	**26.92**
Rekaa	07.28
Tahoma	04.31
Thuluth	17.67
Traditional Arabic	12.87
Zarnew	18.75

Our next goal was to study the recognizer's performance when trained with image samples from multiple typefaces. In this experiment we train our recognizer with computer generated word images from all the eight typefaces together. Thus, a total of 15,432 (1929 × 8) word images are used for training. Once the recognizer was trained, we evaluated the recognizer by recognizing word images from set *d* of the IFN/ENIT database. The evaluation results are shown in Table II. It can be seen from the table that a significant improvement in recognition rate is achieved when we trained the recognizer on multiple typefaces. Thus the variability observed in the training samples due to the different typefaces helps, to some extent, model the variability in human handwriting in the case of Arabic script. A part of the improvement is also due to an eight-fold increase in the training data. To understand the contribution of multiple typefaces alone, we carried one more experiment where we randomly selected only 1929 word images in the eight typefaces for training the recognizer and evaluated the recognizer on set *d*. The recognizer was able to achieve 53.45% WRR which explains that most of the improvement was due to the use of multiple typefaces.

Our next experiment was to use the recognizer trained on multiple typefaces and perform unsupervised HMM adaptation during recognition. We used MLLR regression for parameter adaptation as described in [16], [21]. We experimented with different number of regression classes. The evaluation results on IFN/ENIT set *d* is presented in Table II. It can be seen from the table that significant improvements are achieved using un-supervised HMM adaptation. In the best configuration using 48 regression classes, it leads to improvement in recognition rate by 9.12%, i.e. a reduction in error by one-fourth approximately.

TABLE II: WRR ON SET 'D' OF IFN/ENIT DATABASE USING RECOGNIZER TRAINED ON MULTIPLE TYPEFACES AND USING UNSUPERVIZED ADAPTATION.

System	WRR (%)
All typefaces together	61.35
All typefaces together + Unsupervised adaptation	70.47

Our next set of experiments was related to the idea of using the recognition hypothesis on the test set as training data for the recognizer. To start, we use the recognition hypothesis from the previous step (i.e. multi-typefaces training and unsupervised adaptation during recognition) and use it to generate labels at the character level for each word image of the test set by forced alignment techniques. An interesting aspect to investigate was to compare the results of the recognizer trained on computer generated text on multiple typefaces with the recognizer trained on handwritten text images, but with imperfect labeling (as close to 30% of the word images were wrongly hypothesized). To limit the mislabeled data, we remove the bottom five percent of hypothesized data based on the length-normalized score. After training the recognizer with the hypothesized set *d* of the IFN/ENIT database, we perform recognition on the same set. The evaluation results are presented in Table III. It can be seen from the table that the results are significantly better as compared to the results from the previous approaches. As an extension to this experiment, we use this improved hypothesis to re-label the test set and use it again to train our recognizer. After retraining our recognizer using the improved hypothesis for a few more iterations (until the average length-normalized scores for the hypothesis converges), we evaluate it on the same set. The results are presented in the second row of Table III. We can see from the table that there is a small, but significant, improvement in the recognition rate. In our final set of experiments we use the multi-stream HMMs as presented in [20]. We split the features into two streams such that the computed nine features constitute one stream whereas the derivative features are part of the second stream. Multi-stream HMMs led to a further small, but significant, improvement in recognition rate.

Once we validated our approaches using the set *d* of the IFN/ENIT database, we replicated our experiments on sets *e*, *f*, and *s* of the database without changing the system parameters, i.e. our single typeface, multiple typefaces, and adaptation system was exactly the same as the one used to evaluate set *d*. The only difference was the use of hypothesized data for the corresponding sets as training data. The summary of all the experiments are presented in Table IV.

TABLE III: WRR ON SET 'D' OF IFN/ENIT DATABASE USING HYPOTHESIZED TEST SET FOR TRAINING.

System	WRR (%)
Hypothesized data of the test set used for training	87.16
Hypothesised test-data used for training after five iterations	90.23

From the table we can see that the results, although below the state-of-the-art (c.f. Table II in [20]), are very promising considering that no handwritten data was used for training. We generated the hypothesis on the test set and used it to retrain our recognizer. The area of research seems exciting and needs further investigation and has huge implications as this may greatly reduce, if not completely eliminate, the need for handwritten training sets and its labelling which is a very laborious and time consuming task.

IV. CONCLUSIONS

Handwritten text recognition is an interesting, as well as a challenging research area. Having sufficient labeled data for training a recognizer is one of the prerequisites for good performance. In the paper we present our work on handwritten text recognition in situations where no handwritten training set is available. In one approach, we studied the performance of a recognizer when we train it using computer generated text in single or multiple typefaces. Interesting observations were made on the recognizer's performance on different typefaces when using typefaces of varied degree of visual complexity. Using recognizer trained on text from multiple typefaces performed significantly better than the recognizer trained on text generated using any single typeface. We extended the work by investigating the use of unsupervised HMM adaptation during recognition and found that it can lead to significant improvements in recognition performance. In yet another approach we use the recognizer trained on computer generated text to provide recognition hypothesis on the

TABLE IV: SUMMARY OF THE RESULTS (WRR) FOR HANDWRIITEN TEXT RECOGNIITON ON THE IFN/ENIT DATABASE.

System	The Recognition System Training Data	Evaluation Set			
		d	*e*	*f*	*s*
1	Best individual typeface (Naskh)	26.92	22.10	24.10	27.39
2	Text images from all typefaces together	61.35	55.84	55.14	51.94
3	Text images from all typefaces together + unsupervised adaptation	70.47	66.53	60.93	54.74
4	Test set hypothesised using system 3 used as training data	87.16	84.43	81.44	66.60
5	Five iterations of Test set hypothesised using system 4 used as training data	90.23	87.65	84.78	70.74
6	System 5 + Multi-stream HMMs	**91.61**	**89.61**	**86.58**	**73.11**

handwritten text images of the test set and, in turn, use that hypothesis as labels to train the recognizer. This approach further improved the recognition performance. We achieved good recognition results on a closed vocabulary handwritten word recognition task. The recognition results are below the state-of-the-art on the benchmark database but are reasonably good considering that no handwritten training set was used. More experiments need to be carried out using other handwritten Arabic databases. It will also be interesting to investigate whether the presented approach of initializing a handwritten text recognizer using printed text images can be applied to other scripts, too.

ACKNOWLEDGMENT

The authors would like to acknowledge King Fahd University of Petroleum and Minerals (KFUPM) for supporting this research. The authors would also like to thank Elisa Barney Smith for providing valuable feedback on the work.

REFERENCES

[1] T. Plötz and G. A. Fink, "Markov models for offline handwriting recognition: a survey," *Int. J. Doc. Anal. Recognit.*, vol. 12, no. 4, pp. 269–298, Oct. 2009.

[2] M. T. Parvez and S. A. Mahmoud, "Offline arabic handwritten text recognition: A Survey," *ACM Comput. Surv.*, vol. 45, no. 2, pp. 23:1–23:35, Mar. 2013.

[3] U.-V. Marti and H. Bunke, "The IAM-Database: an English Sentence Database for Offline Handwriting Recognition," *Int. J. Doc. Anal. Recognit.*, vol. 5, no. 1, pp. 39–46, Nov. 2002.

[4] S. A. Mahmoud, I. Ahmad, W. G. Al-Khatib, M. Alshayeb, M. Tanvir Parvez, V. Märgner, and G. A. Fink, "KHATT: An open Arabic offline handwritten text database," *Pattern Recognit.*, vol. 47, no. 3, pp. 1096–1112, Mar. 2014.

[5] M. Pechwitz, S. S. Maddouri, V. Märgner, N. Ellouze, and H. Amiri, "IFN/ENIT - Database of Handwritten Arabic Words," in *7th Colloque International Francophone sur l'Ecrit et le Document , CIFED 2002*, 2002, pp. 129–136.

[6] S. A. Mahmoud, I. Ahmad, M. Alshayeb, and W. G. Al-Khatib, "A Database for Offline Arabic Handwritten Text Recognition," *Image Anal. Recognit.*, pp. 397–406, 2011.

[7] H. S. Baird, "The State of the Art of Document Image Degradation Modelling," in *Digital Document Processing*, B. B. Chaudhuri, Ed. London: Springer London, 2007, pp. 261–279.

[8] T. Varga and B. Horst, "Perturbation Models for Generating Synthetic Training Data in Handwriting Recognition," in *Machine Learning in Document Analysis and Recognition*, vol. 90, S. Marinai and H. Fujisawa, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 333–360.

[9] J. Richarz, S. Vajda, R. Grzeszick, and G. A. Fink, "Semi-supervised learning for character recognition in historical archive documents," *Pattern Recognit.*, vol. 47, no. 3, pp. 1011–1020, Mar. 2014.

[10] Y. Elarian, R. Abdel-Aal, I. Ahmad, M. T. Parvez, and A. Zidouri, "Handwriting synthesis: classifications and techniques," *Int. J. Doc. Anal. Recognit.*, vol. 17, no. 4, pp. 455–469, Oct. 2014.

[11] Y. Elarian, I. Ahmad, S. Awaida, W. G. Al-Khatib, and A. Zidouri, "An Arabic handwriting synthesis system," *Pattern Recognit.*, vol. 48, no. 3, pp. 849–861, Mar. 2015.

[12] H. Miyao and M. Minoru, "Virtual Example Synthesis Based on PCA for Off-Line Handwritten Character Recognition," in *Document Analysis Systems VII*, vol. 3872, H. Bunke and A. L. Spitz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 96–105.

[13] V. Frinken, A. Fischer, H. Bunke, and A. Foornes, "Co-training for Handwritten Word Recognition," in *2011 International Conference on Document Analysis and Recognition (ICDAR)*, 2011, pp. 314–318.

[14] M. Kozielski, M. Nuhn, P. Doetsch, and H. Ney, "Towards Unsupervised Learning for Handwriting Recognition," in *14th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 2014, pp. 549–554.

[15] M. J. F. Gales and P. C. Woodland, "Mean and variance adaptation within the MLLR framework," *Comput. Speech Lang.*, vol. 10, no. 4, pp. 249–264, Oct. 1996.

[16] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Comput. Speech Lang.*, vol. 9, no. 2, pp. 171–185, Apr. 1995.

[17] S. Saleem, H. Cao, K. Subramanian, M. Kamali, R. Prasad, and P. Natarajan, "Improvements in BBN's HMM-based offline Arabic handwriting recognition system," in *Proc. of 10th International Conference on Document Analysis and Recognition, ICDAR'09*, 2009, pp. 773–777.

[18] K. Ait-Mohand, T. Paquet, and N. Ragot, "Combining structure and parameter adaptation of HMMs for printed text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014.

[19] I. Ahmad, L. Rothacker, G. A. Fink, and S. A. Mahmoud, "Novel Sub-character HMM Models for Arabic Text Recognition," in *12th International Conference on Document Analysis and Recognition*, 2013, pp. 658–662.

[20] I. Ahmad, G. A. Fink, and S. A. Mahmoud, "Improvements in Sub-Character HMM Model Based Arabic Text Recognition," in *14th International Conference on Frontiers in Handwriting Recognition*, 2014, pp. 537–542.

[21] S. J. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.2. 1)*. Cambridge University Engineering Department, 2002.

[22] M. Wienecke, G. A. Fink, and G. Sagerer, "Toward automatic video-based whiteboard reading," *Int. J. Doc. Anal. Recognit.*, vol. 7, no. 2, pp. 188–200, 2005.

[23] U.-V. Marti and H. Bunke, "Handwritten sentence recognition," in *Proc. of 15th International Conference on Pattern Recognition. ICPR-2000*, 2000, pp. 463–466.