2009 10th International Conference on Document Analysis and Recognition

A New Arabic Printed Text Image Database and Evaluation Protocols

Fouad SLIMANE[1,2], Rolf INGOLD[1], Slim KANOUN[2], Adel M. ALIMI[2], Jean HENNEBERT[1,3]

[1] *DIVA Group, Department of Informatics, University of Fribourg, Fribourg, Switzerland*
[2] *REsearch Group on Intelligent Machines (REGIM), ENIS, University of Sfax, Sfax, Tunisia*
[3] *Business Information System Institute, HES-SO // Wallis, Sierre, Switzerland*

Fouad.Slimane@unifr.ch, Rolf.Ingold@unifr.ch, Slim.Kanoun@enis.rnu.tn, Adel.Alimi@enis.rnu.tn, Jean.Hennebert@hevs.ch

Abstract

We report on the creation of a database composed of images of Arabic Printed words. The purpose of this database is the large-scale benchmarking of open-vocabulary, multi-font, multi-size and multi-style text recognition systems in Arabic. The challenges that are addressed by the database are in the variability of the sizes, fonts and style used to generate the images. A focus is also given on low-resolution images where anti-aliasing is generating noise on the characters to recognize. The database is synthetically generated using a lexicon of 113'284 words, 10 Arabic fonts, 10 font sizes and 4 font styles. The database contains 45'313'600 single word images totaling to more than 250 million characters. Ground truth annotation is provided for each image. The database is called APTI for Arabic Printed Text Images.

1. Introduction and motivations

With a quite large user base of about 300 million people worldwide, Arabic is important in the culture of many people. In the last fifteen years, most of the efforts in Arabic text recognition have been put for the recognition of scanned off-line printed documents [1][2][3][4]. Most of these developments have been benchmarked on private databases and therefore, the comparison of systems is rather difficult.

To our knowledge, there are currently few large-scale image databases of Arabic printed text available for the scientific community. One of the only references we have found is about the ERIM database containing 750 scanned pages collected from Arabic books and magazines [5]. However, it seems difficult to have access to this database. In the field of Arabic handwriting recognition, public databases do exist such as the freely available IFN/ENIT-database [6] Open competitions are even regularly organized using this database [7][8].

On the other hand, text corpus or lexical databases in Arabic are available from different associations or institutes [9][10][11][12]. However, such text corpora are not directly usable for the benchmarking of recognition systems that take images as input.

Considering this, we have initiated the development of a large database of images of printed Arabic words. This database will be used for our own research and will be made available for the scientific community to evaluate their recognition systems. The database has been named APTI for Arabic Printed Text Image.

The purpose of the APTI database is the large-scale benchmarking of open-vocabulary, multi-font, multi-size and multi-style text recognition systems in Arabic. The images in the database are synthetically generated from a large corpus using automated procedures. The challenges that are addressed by the database are in the variability of the sizes, fonts and style used to generate the images. A focus is also given on low-resolution images where anti-aliasing is generating noise on the characters to recognize. By nature, APTI is well suited for the evaluation of screen-based OCR systems that take as input images extracted from screen captures and images embedded in web pages or pdf documents. Performances of classical scanned-based OCR or camera-based OCR systems could also be measured using APTI. However, such evaluations should take into account the absence of typical artefacts present in scanned or camera documents.

The objective of this paper is to describe the APTI database and the evaluation protocols defined on the database. In section 2, we present details about lexicon, fonts, font-sizes, rendering procedure, Sources of variability and ground truth description. In section 3, statistical information about the content of the database are given.

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.155

The evaluation protocols are showed in section 4. Finally, some conclusions are presented in section 5.

2. Specifications of APTI-Database

2.1 Lexicon

The APTI database contains a mix of decomposable and non-decomposable word images. Decomposable words are generated from root Arabic verbs using Arabic schemes [13] whereas non-decomposable words are formed by Arabic proper names, general names, country/town/village names, Arabic prepositions, etc.

To generate the lexicon, we have parsed different Arabic books such as *The Muqaddimah - An introduction to history of Ibn Khaldun*[1] *and Al-bukhala of Gahiz*[2] as well as a collection of recent Arabic newspapers articles taken from the Internet and a large lexicon file produced by [13]. This parsing procedure totalled 113'284 single different Arabic words, leading to a pretty good coverage of the Arabic words mostly used in texts.

2.2 Fonts, styles and sizes

Taking as input the words in the lexicon, the images of APTI are generated using 10 different fonts presented in Figure 1: Andalus, Arabic Transparent, AdvertisingBold, Diwani Letter, DecoType Thuluth, Simplified Arabic, Tahoma, Traditional Arabic, DecoType Naskh, M Unicode Sara. These fonts have been selected to cover different complexity of shapes of Arabic printed characters, going from simple fonts with no or few overlaps and ligatures (AdvertisingBold) to more complex fonts rich in overlaps, ligatures and flourishes (Diwani Letter).

Figure 1. Fonts used to generate the APTI database: (A)Andalus, (B)Arabic Transparent, (C)AdvertisingBold, (D)Diwani Letter, (E)DecoType Thuluth, (F)Simplified Arabic, (G)Tahoma, (H)Traditional Arabic, (I)DecoType Naskh, (J)M Unicode Sara

[1] Ibn Khaldoun, (May 27,1332 – March 19, 1406) was a famous historien, scholar, theologian, and statesman born in North Africa in presentday Tunisia. (http://en.wikipedia.org/wiki/Ibn_Khaldoun)

[2] Al-Jahiz, (born in Basra, c. 781 – December 868 or January 869) was a famous Arab scholar, believed to have been an Afro-Arab of East African descent.(http://en.wikipedia.org/wiki/Al-Jahiz)

Different font sizes are also used in APTI: **6, 7**, **8**, **9**, **10**, **12**, **14**, **16**, **18** and **24** points. We also used 4 different styles namely plain, italic, bold and combination of italic and bold.

These sizes, fonts and styles are widely used on computer screen, Arabic newspapers and many other documents. The combination of fonts, styles and sizes guaranties a wide variability of images in the database.

2.3 Rendering procedure

The text images are generated using automated procedures. As a consequence, artefacts or noise usually present for scanned or camera-based documents are not present in the images. Such degradations could actually be artificially added, if needed [14], but it is currently out of the scope of APTI.

Image generation of text, for example on screen, can be done in many different ways. They are usually all leading to slight variations of the target image. We have opted for a rendering procedure that allows us to include effects of downsampling and antialiasing. These effects are interesting in terms of variability of the images, especially in low-resolution. An example is given in Figure 2.

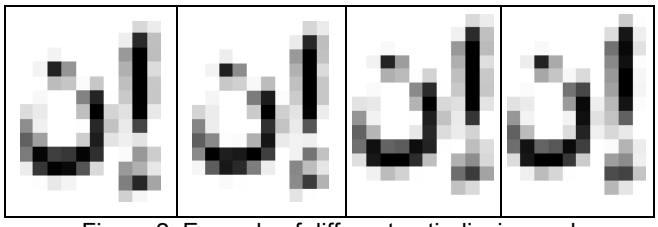

Figure 2. Example of different anti-aliasing and downsampling effects applied to the same source image

The procedure involves the downsampling of a high resolution source image into a low resolution image using antialiasing filtering. We also use different grid alignments to introduce variability in the application of the antialiasing filter. The details of the procedure are the following:

1. A gray-scale source image is generated in high resolution (360 pixels/inch) from the current word in the lexicon, using the selected font, size and style.
2. Columns and rows of white pixels are added to the right hand side and to the top of the image. The number of columns and rows is chosen to have a height and width multiple of the downsampling factor. This effect allows to have the same deformation in all images and artificially moving the downsampling grid.
3. Downsampling and antialiasing filtering are applied to obtain the target image in lower resolution (72 pixels/inch). The target image is in grey level. The downsampling and antialiasing algorithms are the one implemented in the Java class Image. In our implementation, we used the SCALE_SMOOTH option of the Java method which

optimizes the downsampling algorithm selection according to quality and speed.

2.4 Sources of variability

The sources of variability in the generation procedure of text images in APTI are the following:

1. 10 different fonts;
2. 10 different sizes;
3. 4 different styles;
4. Various forms of ligatures and overlaps of characters thanks to the large combination of characters in the lexicon and thanks to the used fonts;
5. Very large vocabulary that allows to test systems on unseen data;
6. Various artefacts of the downsampling and antialiasing filters due to the insertion of columns of white pixels at the beginning of image words;
7. Variability of the height of each word image.

The last point of the previous list is actually intrinsic to the sequence of characters appearing in the word. In APTI, there is actually no a priori knowledge of the baseline position and it is up to the recognition algorithm to compute the baseline, if needed.

2.5 Ground truth description

Each word image in APTI database is fully described using an XML file containing ground truth information about the sequence of characters as well as information about the generation. An example of such XML file is given in Figure 3.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <wordImage id="78">
  - <content transcription="الآلاف" nPaws="4">
      <paw id="1" nbChars="1">Alif_I</paw>
      <paw id="2" nbChars="2">Laam_B TildAboveAlif_E</paw>
      <paw id="3" nbChars="2">Laam_B Alif_E</paw>
      <paw id="4" nbChars="1">Faa_I</paw>
    </content>
    <font name="Arabic Transparent" fontStyle="Plain" size="24" />
    <specs encoding="png" width="96" height="36" effect="none" />
    <generation type="downsampling5" renderer="java" filtering="antialiasing" />
</wordImage>
```

Figure 3. Example of XML file including ground truth information about a given word image

The XML file is composed by four markups sections:

- *Content*: in this element, we have the transcription of Arabic word, the number of Piece of Arabic Word (nPaws) and sub-elements for each PAW with the sequence of characters. In our representation, characters are identified using plain English labels as described below.
- *Font:* in this element, we specify the font name, font style and size used to generate the word image.
- *Specs:* in this element, we present the encoding of image, width, height and eventual additional effect.
- *Generation:* in this element, we indicate the type of generation, the tool used for generation and the used filter in generation. In the current version of APTI, this element is constant as the same generation procedure has been applied. The type 'downsampling5' is here indicating that the generation procedure correspond to a downsampling, using factor 5, from high resolution source images as explained in Section 2.3.

Table 1. Distribution of characters in the different sets

Char label (Char)	Set 1	Set 2	Set 3	Set 4	Set 5	Set 6
Alif (ا)	15078	14925	15165	15120	15046	15019
Baa (ب)	4513	4763	4692	4704	4730	4717
Taaa (ت)	9926	9884	9897	9797	9942	9897
Thaa (ث)	634	633	631	634	643	628
Jiim (ج)	1893	1897	1887	1924	1915	1939
Haaa (ح)	2953	2963	3017	2933	3000	3000
Xaa (خ)	1407	1435	1439	1401	1403	1407
Daal (د)	3187	3033	3075	2990	3028	3086
Thaal (ذ)	514	520	528	504	516	518
Raa (ر)	6304	6243	6169	6335	6253	6267
Zaay (ز)	1064	1054	1054	1066	1042	1045
Siin (س)	3674	3556	3674	3512	3629	3603
Shiin (ش)	1457	1446	1418	1434	1455	1458
Saad (ص)	1374	1377	1388	1411	1371	1389
Daad (ض)	922	943	936	906	921	920
Thaaa (ط)	1419	1426	1431	1426	1446	1462
Taa (ظ)	242	238	240	238	239	241
Ayn (ع)	2764	2823	2769	2718	2755	2723
Ghayn (غ)	981	970	983	984	990	1004
Faa (ف)	2305	2256	2221	2313	2339	2315
Gaaf (ق)	2784	2734	2853	2883	2762	2803
Kaaf (ك)	2101	2090	2099	2145	2136	2140
Laam (ل)	6745	6926	6972	7002	6790	6724
Miim (م)	7871	7836	7957	7806	7797	7817
Nuun (ن)	7484	7433	7289	7316	7400	7264
Haa (ه)	2670	2687	2590	2718	2705	2724
Waaw (و)	4421	4313	4325	4333	4264	4352
Yaa (ي)	6641	6630	6876	6685	6648	6735
NuunChadda (نّ)	225	224	224	223	224	223
YaaChadda (يّ)	725	727	709	719	735	733
Hamza (ء)	192	187	190	193	192	188
HamzaAboveAlif (أ)	1437	1483	1455	1512	1456	1427
HamzaUnderAlif (إ)	253	250	256	247	248	247
TildAboveAlif (آ)	84	84	83	83	83	83
TaaaClosed (ة)	1417	1407	1394	1364	1409	1385
AlifBroken (ى)	162	161	164	163	161	161
HamzaAboveAlifBroken(ئ)	210	208	208	209	208	210
HamzaAboveWaaw (ؤ)	89	90	89	91	89	90
Quantity of characters	***108'122***	***107'855***	***108'347***	***108'042***	***107'970***	***107'944***
Quantity of PAWs	***45'982***	***45'740***	***45'792***	***45'884***	***45'630***	***45'805***
Quantity of words	***18'897***	***18'892***	***18'886***	***18'875***	***18'868***	***18'866***

The different character labels can be observed in Table 1 showing statistics. As the shape of Arabic characters are varying according to their position in the word, the character labels also include a suffix to specify the position of the character in the word: "B" standing for beginning, "M" for Middle, "E" for end and "I" for isolated. The character "Hamza" being always isolated, we don't use the position suffix for this character. We also artificially inserted characters labels such as "NuunChadda" or "YaaChadda" to represent the character shape issued from the combination of "Nuun" and "Chadda" or "Yaa" and "Chadda".

3. Database statistics

The APTI database consists of 113'284 different single words. Table 2 shows the total quantity of word images, PAWs and characters in APTI.

Table 2. Quantity of words, PAWs, characters in APTI

	Nbr of Words	Nbr of PAWs	Nbr of characters
	113'284	274'833	648'280
*10 *Fonts * 10 Font Sizes * 4 Font Styles*			
Total	***45'313'600***	***109'933'200***	***259'312'000***

3.1 Division into sets

We have divided the database into six equilibrated sets to allow for flexibility in the composition of development and evaluation partitions. The words in each set are different but the distribution of all used letters is nearly the same in the various sets (see Table 1). For more details about occurrence of each shape of characters in different sets, we refer to [15]. The five first sets are available for the scientific community and the sixth set is kept internal for potential future evaluation of systems in blind mode.

The algorithm for the distribution of words in the different sets has been designed to have similar allocations of letters and words in all sets. The steps of the algorithms are the following. First, we read all the words from the database and we accumulate the number of occurrence of each letters. The letters are then sorted according to their number of occurrence, from the smallest number of occurrence to the largest. Second, bins (vectors) are created for each letters and they are ordered according to the occurrences computed in step 1. For each word of the database, we go through the bins and we look if the word contains the character associated to the bin. If so, then the word is associated to the bin and we go to the next word. Doing this, we actually build sets of words having letters with low occurrences. Third, we go through each bin and distribute the word sequentially in our final six sets. For more details, we refer to [15].

4. Evaluation Protocols

In this section, we propose the definition of a set of robust benchmarking protocols on top of the APTI database. Preliminary experiments with a baseline recognition system have helped in calibrating and validating these protocols.

4.1 Error estimation

The objective of any benchmarking of recognition systems is to estimate, as reliably as possible, the classification error rate $\hat{P}_e$. It is important to keep in mind that, whatever the task and data used, $\hat{P}_e$ is a function of the split of the data into training and test sets. Different splits will result in different error estimates. Hopefully, APTI is composed of quite large sets of data, which is helping in reaching stable estimates of $\hat{P}_e$.

Our objective is then to obtain a reliable estimate of $\hat{P}_e$ while keeping the computation load tractable. Therefore, we have opted for a *cross validation method*, as described in [16], Section 7]. The idea is to reach a trade-off between the *holdout method* which leads to pessimistic and biased values of the error rate and the *leave-one-out method* that gives a better estimate but at the cost of larger computational requirements. The procedure is to perform independent runs on 5 different partitions between training and testing data.

The final error estimate is taken as the average of the error rates obtained on the different partitions.

$$\hat{P}_e = \frac{1}{5}\sum_{i=1}^{5}\hat{P}_{e,i}$$

In the previous formula, $\hat{P}_{e,i}$ is the error rate obtained independently on a system trained and tested using the sets defined in partition *i*. The procedure actually corresponds to computing the average of performance of 5 independent systems.

4.2 Train and test conditions

Table 3: APTI protocols

Protocol name	Train choice Tr(font, Style, Size)	Test choice Te (font, Style, Size)
APTI 1	Tr(B, p, 10)	Te(B, p, 10)
APTI 2	Tr(B, p, 10)	Te(B, i, 10)
APTI 3	Tr(B, p, 10)	Te(B, b, 10)
APTI 4	Tr(B, p, 10)	Te(B, bi, 10)
APTI 5	Tr(B, p, [6, 10, 14, 18])	Te(B, p, [6, 10, 14, 18])
APTI 6	Tr(B,[p,i,b], [6, 10, 14, 18])	Te(B,[p,i,b], [6, 10, 14, 18])
APTI 7	Tr([A,B,C,F,H], p, 10)	Te([A,B,C,F,H], p, 10)
APTI 8	Tr([D,E,G,I,J], p, 10)	Te([D,E,G,I,J], p, 10)
APTI 9	Tr([A,B,C,F,H], [p,i,b], 10)	Te([A,B,C,F,H], [p,i,b], 10)
APTI 10	Tr([D,E,G,I,J], [p,i,b], 10)	Te([D,E,G,I,J], [p,i,b], 10)
APTI 11	Tr([A,B,C], p, 10)	Te([F,H], p, 10)
APTI 12	Tr([D,E,G], p, 10)	Te([I,J],i, 10)
APTI 13	Tr([A,B,C], p,[6,10,14,18])	Te([F,H], p, [6,10,14,18])
APTI 14	Tr([D,E,G], p,[6,10,14,18])	Te([I,J], p, [6,10,14,18])
APTI 15	Tr(B, p, 6)	Te(B, p, 6)
APTI 16	Tr(B, p, 8)	Te(B, p, 8)
APTI 17	Tr(B, p, 10)	Te(B, p, 6)
APTI 18	Tr(B, p, 6)	Te(B, p, 10)
APTI 19	Tr(B, p, [6, 10, 14, 18])	Te(B, p, [7,9,12,24])
APTI 20	Tr(all, all, all)	Te(all, all, all)

Using the procedure described in section 4.1, we can define different combinations of train and test conditions. The objectives are to measure the impact of some of the variability of the data. We therefore propose 20 protocols as summarized in Table 3. The notations Tr(font, style,

size) and Te(font, style, size) define the training and testing conditions with:

1. the font label as indicated in Figure 1
2. the style where *p*, *i*, *b* and *bi* are for plain, italic, bold and bold+italic
3. the size in points

We suggest researchers willing to define new protocols to use this notation to specify the conditions of their training and testing.

The objectives behind the protocols of Table 3 can be explained as follows:

- **APTI 1**: This is the baseline protocol where performances should be the highest as there are no mismatched between training and testing conditions.
- **APTI 2,3,4**: We measure here the capability of systems trained using plain style to generalize on italic, bold and bold+italic.
- **APTI 5,6**: While using the same font, we measure the capability of the system to treat different sizes.
- **APTI 7,8,9,10**: These experiments measure the capability of systems to recognize muti-font text.
- **APTI 11,12,13,14**: We measure the capability of systems to recognize unseen fonts text.
- **APTI 1,15,16,17,18,19**: Firstly, we measure the potential degradation of performance using smaller sizes. Secondly, we measure the capability to recognize unseen sizes.
- **APTI 20**: This is the global experiment where all available data is used for training and testing.

5. Conclusion

APTI, a new large Arabic printed text images database is presented together with evaluation protocols. APTI aims at the large-scale benchmarking of open-vocabulary text recognition systems. While it can be used for the evaluation of any OCR systems, APTI is, by nature, well suited for the evaluation of screen-based OCR systems. The challenges addressed by the database are in the variability of the sizes, fonts and style and the protocols that are defined are crafted to put into evidence the impact of such variability. APTI will be made publicly available for the purpose of research.

6. References

[1] M. S. Khorsheed, "Offline recognition of omnifont Arabic text using the HMM ToolKit (HTK)". Pattern Recognition Letters, Vol 28, 2007, pp. 1563 - 1571

[2] H. A. Al-Muhtaseb, Sabri A. Mahmoud, Rami S. Qahwaji, "Recognition of off-line printed Arabic text using Hidden Markov Models". European Signal Processing Conference, Lausanne, Switzerland, 2008, Vol. 88, Issue 12, pp. 2902 - 2912

[3] Z. Shaaban, "A New Recognition Scheme for Machine-Printed Arabic Texts based on Neural Networks". Proceedings of World Academy of Science, Engineering and Technology, Vol. 31, Vienna, Austria, 2008

[4] F. Slimane, R. Ingold, M. A. Alimi and J. Hennebert, "Duration Models for Arabic Text Recognition using Hidden Markov Models". CIMCA 2008, Vienne, Austria, 2008

[5] S. Schlosser, "ERIM Arabic Database", Document Processing Research Program, Information and Materials Applications Laboratory, Environmental Research Institute of Michigan, 1995

[6] M. Pechwitz, S. S. Maddouri, V. Maergner, N. Ellouze, and H. Amiri, "IFN/ENIT - database of handwritten Arabic words". In Proc. of CIFED 2002, Hammamet, Tunisia, October 21-23 2002, pp. 129–136

[7] V. Margner, M. Pechwitz, H. El Abed, "Arabic Handwriting Recognition Competition", In ICDAR, 2005, pp.70 - 74

[8] V. Margner and H. E. Abed. "ICDAR 2007 Arabic handwriting recognition competition". In ICDAR, Sept. 2007 vol. 2, pp. 1274–1278.

[9] D. Graff, K. Chen, J. Kong, and K. Maeda, "Arabic Gigaword Second Edition", Linguistic Data Consortium, Philadelphia, 2006

[10] R. Abbes, J.D. Hassoun, "The Architecture of a Standard Arabic Lexical Database, Some Figures, Ratios and Categories from the DIINAR.1 Source Program", Workshop of Computational Approaches to Arabic Script-Based Languages, Geneva, 2004

[11] A. AbdelRaouf, C. A Higgins, and M. Khalil, "A Database for Arabic Printed Character Recognition". ICIAR 2008, LNCS 5112, pp. 567 – 578

[12] C. LaPre, Y. Zhao, C. Raphael, R. Schwartz, J. Makhoul, "Multi-font Recognition of printed arabic using the BBN Byblos speech recognition system", ICASSP.1996, vol. 4 pp.2136-2139

[13] S. Kanoun, A. M. Alimi, Y. Lecourtier, "Affixal approach for Arabic decomposable vocabulary recognition a validation on printed word in only one font", In ICDAR, Sept. 2005, vol. 2 pp.1025 - 1029

[14] H. S. Baird. "State of the Art of Document Image Degradation Modeling". Proceedings of the 4th IAPR Workshop on Document Analysis Systems, DAS 2000.

[15] F. Slimane, R. Ingold, S. Kanoun, M. A. Alimi and J. Hennebert, "Database and Evaluation Protocols for Arabic Printed Text Recognition". Internal Publication, DIUF, University of Fribourg, Switzerland, 2009

[16] A. K. Jain, R. Duin and J. Mao, "Statistical Pattern Recognition: A Review", IEEE Trans. on Pattern Analysis and Machine Intelligence, January 2000, Vol. 22, No. 1