Automatic Script and Type Identification in Bi-lingual Forms
Afef Kacem Echi and Asma Saïdani
Pages 25 – 32
DOI: http://dx.doi.org/10.21700/ijcis.2016.104

Automatic Script and Type Identification in Bi-lingual Forms

Afef Kacem Echi Asma Saïdani
LaTICE Laboratory, University of Tunis
Avenue Taha Hussein Montfleury, 1008 Tunis, TUNISIA
Afef.kacem@esstt.rnu.tn, saidaniasma@yahoo.fr

Abstract: *In this paper we have developed a system that can automatically discriminate between machine-printed and handwritten words in structured bi-lingual (Arabic and French) form document layout. Our system has been applied in the context of Tunisian National Health Insurance Fund for medical care costs refund with encouraging results. In the used forms, handwritten data usually touch or cross the preprinted form frames and texts, creating complex problems for the recognition routines. Each text type should also be processed using different methods in order to optimize the recognition accuracy. This work aims to address these issues and to especially solve the problem of machine-printed/handwritten and Arabic/French word discrimination. To this end, we computed co-occurrence matrix of oriented gradients from word's image and used it as input to a k-Nearest Neighbor classifier. Experiments are carried on 20 forms. An average script identification rate of 98.31% is achieved.*

Keywords: *Bi-lingual Forms, Word Script and Type Identification, Text-line Segmentation, Word Extraction, Co-occurrence Matrix of Oriented Gradients, Classification.*

Received: June 10, 2016 | ***Revised:*** *June* 25, 2016 | ***Accepted:*** August 31, 2016

1. Introduction

The automation of form processing is attracting intensive research interests due to its wide application and its reduction of the heavy workload due to manual processing. In fact, great number of applications uses documents presenting printed text and handwriting. Old documents, petitions, requests, applications for college admission, letters, requirements, memorandums, envelopes and bank checks are some examples. A considerable obstacle to optical character recognition (OCR) systems is the mixture of printed and handwritten text in the same image. Each text type should be processed using different methods in order to optimize the recognition accuracy.

This paper deals with the essential concepts of form analysis and recognition. It specially concerns the acquirement sheets needed for adherent subscription in Tunisian National Health Insurance Fund (CNAM: "Caisse Nationale de l'Assurance Maladie"). The acquirement sheet is used for data collection, with fields designed for this purpose. It is composed of printed and fixed fields and answer fields to be filled by the adherent. The fixed fields are intended to identify the fund, to inform the adherent or to question him. The fields are grouped into blocks. Blocks are clearly separated by frames (see Fig. 1). With the proposed system, the adherent information sheets can be automatically analyzed and data captured from sheet fields.

To distinguish between machine-printed and handwritten texts, various classification techniques were previously proposed: neural networks [1], linear polynomial for discrimination function [2], Fisher [3] and tree classifiers [4], Hidden Markov Model [5] or minimal distance classifiers [6]. Others works addressed the problem of identifying Arabic and Latin scripts by various features and classifiers. In [8] Haboubi and al. used Gabor filters, gray-level co-occurrence matrices and wavelets separately to discriminate between Arabic and Latin machine-printed words. Benjelil and al. proposed in [7] an identification system based on steerable pyramid transform. Mezghani and al. considered in [9] affine moment invariants, the number and the XY position of the top and the bottom extrema, the maximal amplitude obtained from the difference between the top and the bottom profiles for Arabic/French and Handwritten/Printed words identification. In [10], Benjelil and al. present a performance comparison of curvelets, dual-tree complex wavelet and discrete wavelet transform in handwritten words classification (Arabic and Latin).

In our former works [11] and [13], we successfully employed different sets of features such as word vertical projection variance, baseline profile, run-length and crossing count histograms, bottom diacritic, loop position and elongate descenders. But these structural features are script dependent and frequency domain features (Gabor filters, wavelets transform, etc. which are not script dependent) are not proficient to work with small size images like word images. Recently, we respectively investigate the use of black run lengths based features and pyramid histogram of oriented gradients, to exploit the writing orientation and we achieved satisfactory results, in [15] [14]. But, we noted that including co-occurrence with various positional offsets, the features descriptors can express complex shapes of writing with local and global distributions of gradient orientations. That is why we proposed in recent work [12], the use of co-occurrence matrix of oriented gradients. This feature was applied on isolated words and this work is an attempt to use it for an automatic script and type identification in bilingual forms. Note that when the identification is performed by words and not by line, it is possible to analyze more complex forms which mix in the same line both type and script of characters.

This paper is organized as follows: In section 2 each step of the overall system is presented. Section 3 considers the training, tests and results. Finally, section 4 summarizes the conclusions and future improvements.

Figure 1. A sample of a used database

2. Proposed System

Fig. 2 shows an overview of the proposed system: the applied image preprocessing techniques, the segmentation of the text into text-lines and words, the extracted features from word's images and the classification process to separate between machine-printed/handwritten and Arabic/French words in the used application forms.

The handled forms are characterized by a wide range of writings variations due to the multiplicity of the writing styles and writers. Their structure is fixed and contains printed and handwritten text, graphics (logos and signature), checkboxes, dotted reference lines, date field insertion, etc. Fig. 1 shows an example of possible images to be processed.

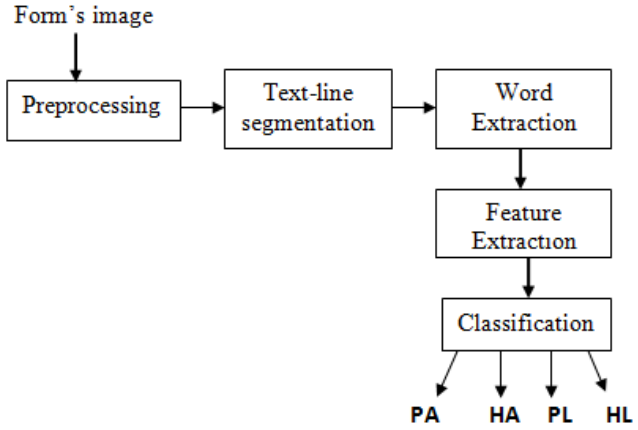

Figure 2. Overview of the proposed system

2.1 Pre-processing

The form preprocessing prepares the acquired image to text-line segmentation and word extraction. Tow operations are accomplished in this step. A 3_3 median filter is firstly applied to decrease the noise in the image. Then, the text is separated from background using the Otsu bi-level method [16].

2.2 Segmentation into text-lines

Although in printed documents, text-line extraction is a rather straightforward process, in the case of handwritten or mixed documents there exist several challenges. One of the problems making this task crucial is the presence of touching or overlapping components where neighboring letters or words touch each other's or cross the preprinted form frames and texts. These touching components are generally due to presence of noise in images, writing style and narrow spacing interlines. This problem is common in unconstrained Arabic manuscripts since most of its letters are ascendant or descendant and contain special marks and dots.

The proposed text-line extraction method for forms strives towards dealing with all aforementioned challenges. It comprises two main steps: 1) connected component extraction and their partition based on the average character height and width and a rule-based classification and 2) text-lines extraction by Run

Length Smoothing Algorithm (RLSA), horizontal projection, segmentation of vertically touching components and their assignment to their respective text-lines.

Connected Component Extraction and Partition: We start by extracting the connected components from the form's binary image. As it is common to have components describing one character, multiple characters, a whole word, accents, and characters from adjacent touching text-lines, the connected components set contains components of a different profile with respect to width and height. For that reason, we calculate the bounding box coordinates for each connected component, the average character height AH and the average character width AW in the whole document image. We then divide the connected component set into different sub-sets in order to deal with these categories separately. More precisely, we separate the connected components domain into three distinct subsets: "Subset 1", "Subset 2", and "Subset 3" as done by [17].

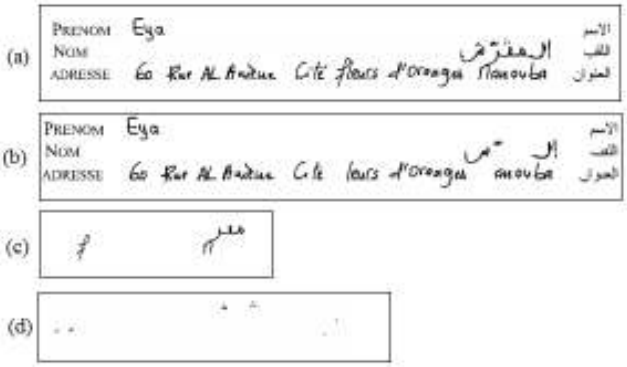

Figure 3. Examples of connected components from (a) that are assigned to: (b) "Subset 1", (c) "Subset 2", and (d) "Subset 3"

As shown in Fig. 3(b), "Subset 1" includes all components which correspond to the majority of the characters having a height range: (0.5*AH ≤ H<3*AH) and (0.5*AW≤W) where H and W respectively denote the component's height and width, respectively. "Subset 1" is used to exclude accents and components that are large in height and belong to more than one text-line. As it is displayed in Fig. 3(b), connected components that contain ascenders and/or descenders will be included in this sub-set.

In "Subset 2" all large connected components are included (see Fig. 3(c)). Large components are either capital letters or characters from adjacent touching text-lines. The height of these components is defined as follows: H≥3*AH. "Subset 2" is used to grasp all connected components that exist due to touching text-lines. We assume that the corresponding height will exceed three times the average character height.

Characters as accents, punctuation marks and small characters should belong to "Subset 3" (see Fig. 3(d)). These components are described by ((H <3*AH) and (W <0.5*AW)) or ((H<0.5*AH) and (W > 0.5*AW)). "Subset 3" is defined since accents usually have width less than half the average character width or height less than half the average character height.

Text-line Extraction: This step includes:

- The application of RLSA on the image containing the connected components of the "subset 1".
- The horizontal profile projection of the resulting image.
- The profile smoothing which aims to alleviate the presence of local minima.
- The estimation of minimum from the smoothed profile horizontal projection. Each minimum in the profile is considered as a separator between the text-lines (See Fig. 4).

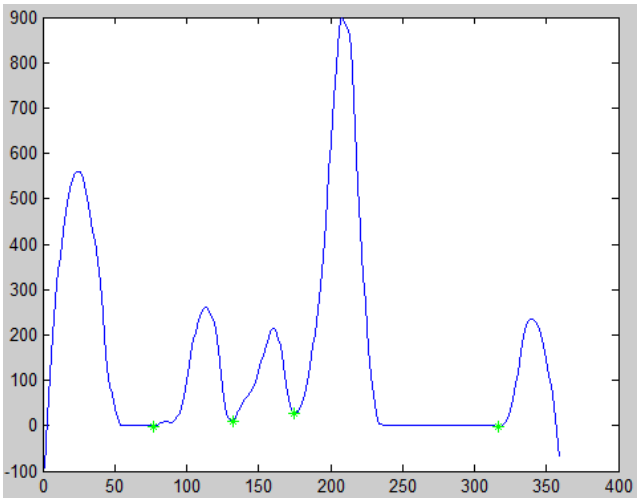

Figure 4. Minimum detection from the smoothed profile horizontal projection of image in Fig. 3(b)

These separators are potential points for text-line extraction. The minimum tracing represents each minimum by a horizontal line. These lines correspond to separators of existing text-lines (see Fig. 5).

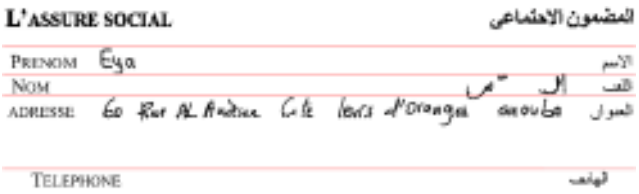

Figure 5. The minimum tracing

- Detection and removal of the connected components in the intersection with separators lines. These components will be assigned to a new sub-set, called "subset 4" in order to reallocate them to their respective text-lines. At this level, we are able to estimate the number of text-lines in the image. Each text-line is characterized by the coordinates of its lower and higher ordinates.
- Assignment of connected components of "subset 4" to their text-lines: For each component of "subset4", having a single intersection with separator line, we first must find the two adjacent text-lines that involve it: L_i and L_{i+1}. We then extract its contour to

locate the intersection points between the touching component and the separator line. Afterward, we cut the component's contour into two component parts up and down and we calculate their respective pixel numbers. We finally compare between the numbers of pixels and assign the component to its appropriate text-line. At this level, we have to secondly update all text-line components (see Fig. 6).

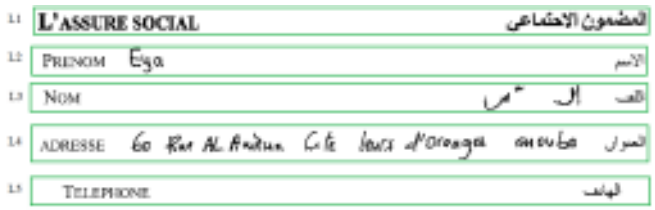

Figure 6. The first text-line's components updating

- Restitution of connected components of "subset 3" to the original image: Recall that connected components of "subset 3" correspond to accents, diacritic, punctuation marks and small characters. Such components can be in two positions: inside the bounds of a text-line or at the intersection with a separator line. The first case allows direct assignment of these components to the current text-line. The second case, leads us to treat them as components of "subset 4" where we must assign them to their closest text-line. Fig. 8 represents the third text-line's components updating after the "subset 3"'s components assignment process.
- Restitution of "subset 2"'s connected components to the original image (see Fig. 7).

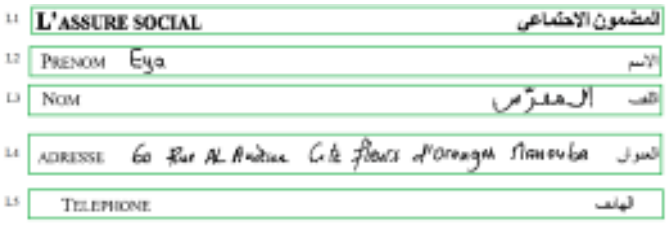

Figure 7. The second text-line's components updating

For these components with generally correspond to vertically connected letters, belonging to two adjacent text-lines, we used the method proposed by [18] to segment them. The idea is to approximate the touching component to models, stored in a code book with their known segmentation: part A and part B composing respectively the first and the second letter. Then, we used these parts to segment the touching component. Thus, there are two steps before touching components can be segmented: 1) a recognition step to find its most similar model, using the shape context descriptor and 2) an approximation step to estimate the transformation aligning the found model to the touching component. Once the touching component is aligned to its most similar model, we used and adjusted the midpoints of the model's parts to segment it. Fig. 8 represents the final result of text-line extraction.

Figure 8. The final result of text-line extraction

2.3 Segmentation into words

For each extracted text-line, we proceed as follows:

- The baseline extraction, using the horizontal projection method,
- The removal of diacritic dots. In fact, punctuation may disturb the inter-words gap which becomes smaller and therefore mislead intra-word gap to inter-word gap. Hence, punctuation is a big problem and therefore it causes many errors in accurate word segmentation.
- The baseband or central band extraction based on the text-line's horizontal projection's profile.
- Connected component extraction, as previously explained. Note that for each connected component, we only considered the intersection zone between it and the baseband to discard letter's extensions (ascender and descender parts) which generally cause touching and overlapping components (see Fig. 9(a)).
- Distance computation. We calculated the Euclidean distance between all pair of adjacent connected components' pixels and retained the smallest distance as effective gap between these components (see Fig. 9(a)).
- Gap classification: We defined a function f that maps a gap in a text-line's image to: 1) Between Word Gap (BWG) or 2) Within Word Gap (WWG). If the set of gaps is g_1, g_2,...,g_n, the function maps some g_i to the WWG and the others to the BWG. For that, we used the K-means cluster with K = 2 (two classes: BWG and WWG as shown in Fig. 9(a) and (b)). Fig. 9(c) displays the obtained results from word extraction.

2.4 Machine-printed/handwritten and Arabic/French Word Discrimination

The proposed method separates machine-printed and handwritten Arabic from French words based on Co-occurrence Matrix of Oriented Gradients (Co-MOG). It is about exploiting the writing orientation as a discriminative descriptor for Arabic and French discrimination. For this, we rely on the observation that we made in the examination of the morphology of these two scripts. Letters in Arabic words, especially of handwritten type or italic machine-printed and as

written from right to left, are generally tilted to the left, following the writing direction (see Fig. 10(a)). In contrast, Letters in French script, especially of handwritten type or in italic machine-printed and as written from left to right, tend to be inclined to the right (see Fig. 10(b)). Thus, Arabic letter strokes are generally diagonally down whereas those written in French are diagonally up. Furthermore, machine-printed Arabic words are characterized by the use of horizontal ligatures, more or less long depended on the used font (see Fig. 10(c)). Oppositely, machine-printed French words are composed by successive letters without any ligature between them (see Fig. 10(d)). Consequently, horizontal strokes would be more frequent in Arabic words than in French words. Note that both scripts use vertical strokes for ascenders. It is those observations that motivated us to explore how the spatial distribution of shape can benefit script and its type identification.

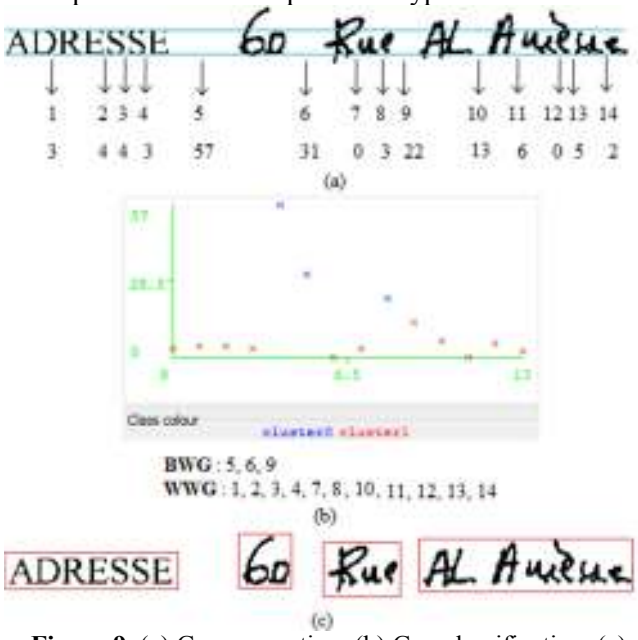

Figure 9. (a) Gap computing, (b) Gap classification, (c) Word extraction result

Figure 10. Machine-printed/Handwritten and Arabic/Latin word identification based on the writing orientation

To achieve that, we proposed a script identification system, based on Co-MOG as it is a shape-based descriptor and we used the k-Nearest Neighbor (k-NN) as classifier for machine-printed/handwritten and Arabic/Latin discrimination at word level. Note that k-NN is the extension of the Nearest Neighbor classifier: An unknown word is classified by assigning it the label most frequently represented among the k nearest word samples. A decision is made by examining the labels of the k nearest neighbors and taking a vote. In this work, we used 1-NN classifier and the Euclidian metric to determine the neighboring words. In Fig. 11, we display the overview of the proposed system where PA (Printed Arabic), PL (Printed Latin), HL (Handwritten Latin) and HA (Handwritten Arabic) are the four classes to which a word may belong.

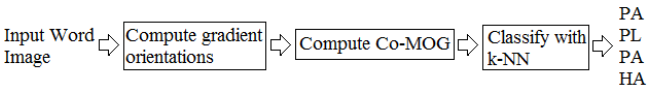

Figure 11. The proposed system overview

Co-MOG is used to express the distribution of gradient information over an image. It captures spatial information by counting the frequency of co-occurrences of oriented gradients between pairs of pixels. The relative locations are reflected by the offset between two pixels as shown in Fig. 12(a). The offset (Δx, Δy) specifies the distance between the pixel of interest and its neighbor. The yellow pixel in the center is the pixel under study and the neighboring blue ones are pixels with different offsets. Each neighboring pixel in blue color forms an orientation pair with the center yellow pixel and accordingly votes to the co-occurrence matrix as illustrated in Fig. 12(b). The frequency of the co-occurrences of oriented gradients is captured at each offset via a co-occurrence matrix as shown in Fig. 12(b).

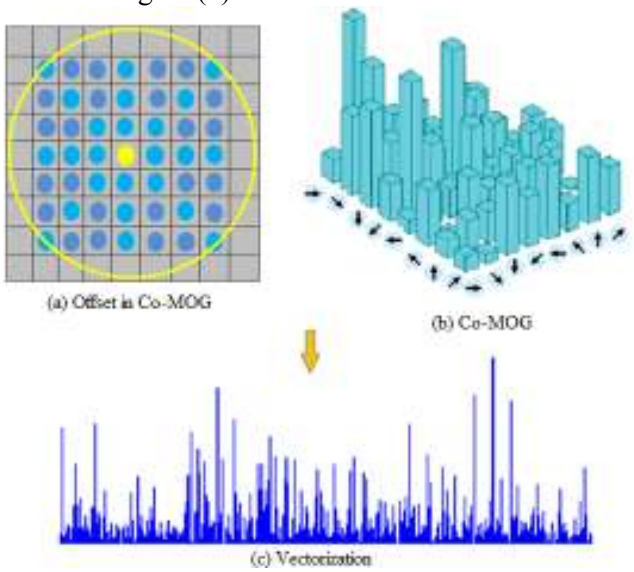

Figure 12. (a) Offset in Co-MOG, (b) Co-occurrence of a word image at a given offset, (c) Vectorization of co-occurrence matrix [19]

Let I be an image of size N*M. We first compute its gradient orientation matrix: R_o as follows:

$$R_o = \arctan\frac{R_y}{R_x}$$

where R_x and R_y denote the horizontal and vertical gradients of the image are computed by applying the 1D centered point discrete derivative masks with the following filter kernels: D_x =(-1 0 1) and $D_y = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$

We label each pixel with one of n discrete orientations. For example, when the number of bin is equal to 8, each pixel takes an orientation in the range [1, 8]. We then compute the Co-MOG matrix, noted P, as follows:

$$P(i,j) = \sum_{x=1}^{N}\sum_{y=1}^{M} \begin{matrix} 1; \text{ if } R_o(x,y)=i \text{ and } R_o(x+\Delta x, y+\Delta y)=j \\ 0; \text{ otherwise} \end{matrix}$$

Note that P is a square matrix and its dimension is *n*n* where *n* is the number of orientation bins. It is defined according to the variation of the offsets that defined the orientation. As Co-MOG describes shapes in details, it is high-dimensional descriptor: a total of 324 features are extracted from each word's image. As some features are likely irrelevant and redundant, we used a Genetic Algorithm (GA) and Principal Component Analysis (PCA) for reducing the dimensionality of the feature vectors.

Note that GA is an efficient method for function minimization. In descriptor selection context, the prediction error of the model built upon a set of features is optimized. The GA mimics the natural evolution by modeling a dynamic population of solutions. The members of the population, referred to as chromosomes, encode the selected features [24]. The goal of PCA is to derive a smaller set of features which accurately represent the original dataset. In particular, PCA finds the linear subspace of lower dimensionality that maximizes the variance of the original set, which is called principal subspace. A comprehensive description of this method can be found in [25].

When applying PCA, the number of features is reduced from 324 to 161 and to 248 if GA is used. More details about the entire identification system can be found in [12]. Fig. 13 shows some obtained results.

Figure 13. Word script and type identification: Machine-printed French words in yellow, Handwritten French words in green, Machine-printed Arabic words in red and Handwritten Arabic words in blue

3. Carried Experiments

We carried out recognition experiments on a variety of forms: about 20 forms containing an average of 157 machine-printed words. Empty forms where obtained from the CNAM and were willingly filled by some adherents. Forms are scanned at a resolution of 600dpi and stored in the format bitmap. To evaluate the proposed word extraction method, we used some samples of form. We then compared between the numbers of extracted words with the total number of words that should be extracted and computed the recall *R* as follows: Number of extracted words Total number of words to be extracted (see Table 1).

Table 1. Word Extraction Evaluation

Form	Number of words to be extracted	Number of extracted words	Recall(%)
$Form_1$	168	162	96.42
$Form_2$	174	158	90.80
$Form_3$	169	154	91.12
$Form_4$	170	157	92.35
$Form_5$	171	155	90.64
$Form_6$	166	154	92.77
$Form_7$	173	157	90.75
$Form_8$	178	161	90.44
$Form_9$	175	162	92.57
$Form_{10}$	171	157	91.81
$Form_{11}$	182	163	89.56
$Form_{12}$	172	157	91.27
$Form_{13}$	175	160	91.42
$Form_{14}$	169	155	91.71
$Form_{15}$	164	155	94.51
$Form_{16}$	177	164	92.65
$Form_{17}$	169	153	90.53
$Form_{18}$	167	153	91.61
$Form_{19}$	181	162	89.50
$Form_{20}$	179	167	93.29
			Average 91.60

To evaluate the proposed word script and type identification method, we used 24000 words as training set of the nearest neighbor classifier. These words are extracted from four public databases that can be accessible publicly: IAM database for Latin handwritten [20], IFN-ENIT [21] and AHTID/MW [22] for Arabic handwritten and APTI [23] for Arabic and Latin printed words with equal number of Printed Arabic (PA), Handwritten Arabic (HA), Printed Latin (PL) and handwritten Latin (HL) words. These sets are composed of multi-fonts words written in ten fonts for Arabic (Andalus, Arabic Transparent, AdvertisingBold, Diwani Letter, DecoType Thuluth, DecoType Naskh, Tahoma, Traditional Arabic, Simplified Arabic and M Unicode Sara) and ten fonts for French (Arial, Monotype Corsiva, ComicSansMS, Edwardian Script ITC, Times New Roman, French Script MT, Impact, Georgia, Arial Black and Tahoma). Note that these fonts cover various complexities of shapes. Fig. 14 shows some words extracted from the used databases.

For each form, the extracted words are used as input to the classifier which returns their scripts and types. We found that in 98.31% of cases the word's script and type is correctly identified (See Table 2).

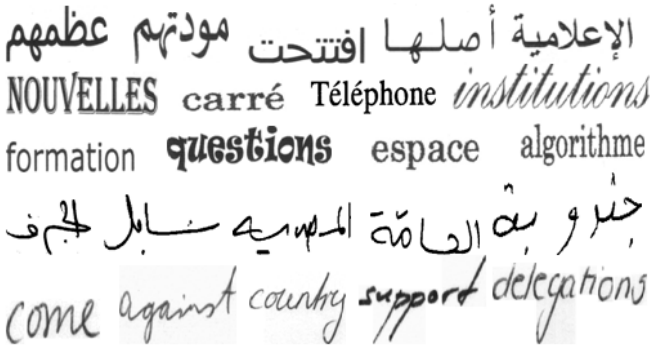

Figure 14. Samples of words extracted from standard Databases.

Table 2: Word Script and Type Identification Evaluation

Form	Word's Number	Accuracy(%)
$Form_1$	162	96.42
$Form_2$	158	98.73
$Form_3$	154	98.05
$Form_4$	157	99.72
$Form_5$	155	98.06
$Form_6$	154	98.70
$Form_7$	157	98.72
$Form_8$	161	98.75
$Form_9$	162	97.53
$Form_{10}$	157	98.72
$Form_{11}$	163	98.15
$Form_{12}$	157	98.08
$Form_{13}$	160	98.12
$Form_{14}$	155	98.06
$Form_{15}$	155	98.70
$Form_{16}$	164	98.78
$Form_{17}$	153	98.69
$Form_{18}$	153	98.03
$Form_{19}$	162	98.14
$Form_{20}$	167	98.20
		Average 98.31

4. Conclusion and future work

In this work, we proposed a script identification system that can automatically discriminate between machine-printed and handwritten words in structured bi-lingual form document layout. The forms are written in English and in French. We assessed the performance of our system on several health-based documents. The obtained results are promising. Several pre-processing techniques have been used to enhance the system accuracy: 1) a first method to segment the form's image into text-lines taking into account the problem of vertically connected text-lines, 2) a second method to segment text-lines into words based on efficient distinction of inter and intra-word gaps using the Euclidian distance and an unsupervised cluster: K-means and 3) a third method to discriminate between machine-printed and handwritten, Arabic and French words by the use of a specific matrix (co-occurrence matrix of oriented gradients) features and the nearest neighbor classifier. From our experimental results, it is shown that the proposed methods achieve satisfactory results: The script identification system accuracy reached 98.31% on a set of 20 forms. In the future, we plan to enhance the proposed text-line and word segmentation methods and to assess the system on a larger set of forms.

First, we have identified several non-functional concerns with respect to pervasive services and then discussed how a component-based methodology can alleviate these concerns. We formalized the component methodology by creating a meta-model of all software aspects of components to allow automated processing of the functional and non-functional properties of components and services. We discussed how context-awareness can be used for composition of component-based services using our previous work on context modelling [16], specification of pervasive services [14] and context management support [15]. We illustrated the core ideas of our infrastructure support by means of an example, and gave an outline of all the steps in the procedure to achieve a customized service. We evaluated our work and can conclude that for targeting a specific platform and incorporating user preferences, our automated context-driven composition infrastructure is able to provide the necessary support for these non-functional concerns typical for pervasive services.

However, more work should be carried out to provide support for cross-cutting non-functional concerns, such as security. Security requirements cannot be derived from component descriptions as is the case for resource requirements. Future work in the short term will focus on how to integrate multi-party negotiation for selecting and instantiating components in the presence of dependencies between component deployments on different devices.

References

[1] F. Farooq, K. Sridharan, and V. Govindaraju, Identifying Handwritten Text in Mixed Documents, ICPR 2006, 18th International Conference on Pattern Recognition, v. 2, pp. 1142 - 1145, 2006.

[2] J. Franke, and M. Oberlander, Writing Style Detection by Statistical Combination of Classifiers in Form Reader Applications, Proceedings of the 2nd Intern. Conference on Document Analysis and Recognition, pp. 581 - 584, 1993

[3] Y. Zheng, H. Li, and D. Doermann, Machine Printed Text and Handwriting Identification in Noisy Document Images, IEEE Transactions on Pattern Analysis and Machine Intelligence, v. 26, n 3, pp. 337 - 353, 2004.

[4] U. Pal, and B. B. Chaudhuri, Machine-printed and Handwritten Text Line Identification, Pattern Recognition Letters, v. 22, n 3 - 4, pp. 431 - 441, 2001.

[5] J. K. Guo, and M. Y. Ma, Separating Handwritten Material from Machine Printed Text Using Hidden Markov Models, Proceedings of the 6th International Conference on Document Analysis and Recognition, pp. 439 - 443, 2001.

[6] E. Kavallieratou, S. Stamatatos, and H. Antonopoulou, Machine-Printed from Handwritten Text Discrimination, Proceedings of the 9th Intern. Workshop on Frontiers in Handwriting Recognition, 26-29 Oct., pp. 312 - 316, 2004.

[7] M. Benjelil, R. Mullot and M. A. Alimi, Language and script identification based on Steerable Pyramid Features, Proceedings of ICFHR, pp. 712-717, 2012.

[8] S. Haboubi, S. Maddouri and H. Amiri, Discrimination between Arabic and Latin from bilingual documents, Proceedings of CCCA, 2011.

[9] A. Mezghani, F. Slimane, S. Kanoun and V. Märgner, Printed/handwritten Arabic Script Identification using Local features and GMMs, Proceedings of of CIFED, 2014.

[10] M. Benjelil and R. Mullot, Performance of curvelets, dualtree complex wavelet and discrete wavelet transform in handwritten word classification, Proceedings of SoCPaR, pp. 53-58, 2014.

[11] A. Saïdani, A. Kacem and A. Belaïd, Identification of machine-printed and handwritten words in Arabic and Latin scripts, Proceedings of ICDAR, pp. 798-802, 2013.

[12] A. Saïdani, A. Kacem and A. Belaïd, Co-occurrence Matrix of Oriented Gradients for Word Script and Nature Identification, Proceedings of ICDAR, 2015.

[13] A. Saïdani, A. Kacem and A. Belaïd, How to separate between Machine-Printed/Handwritten and Arabic/Latin Words?, ELCVIA, Vol. 13, Num. 1, pp. 1-16, 2014.

[14] A. Saïdani and A. Kacem, Pyramid Histogram of Oriented Gradient for Machine-printed/Handwritten and Arabic/Latin word discrimination, Proceedings of SoCPaR, pp. 267-272, 2014.

[15] A. Kacem and A. Sa¨ıdani, A Texture-based Approach for Word Script and Nature Identification, Pattern Analysis and Application (PAA), 2016.

[16] N. Otsu, A Threshold Selection Method from Gray-Level Histograms, IEEE Transactions on Systems, Man and Cybernetics, Volume 9, Num 1, pp. 62-66, 1979.

[17] G. Louloudis, B. Gatos, I. Pratikakis, C. Halatsis, Text-line and word segmentation of handwritten documents, Pattern Recognition, vol. 42, No. 12, pp. 3169-3183, 2009.

[18] N. Aouadi, A. Kacem Echi and A. Belaïd, A Recognition based approach for segmenting touching components in Arabic manuscripts, Proccedings of the 13th Int Conference Document Analysis and Recognition, pp. 21-25, 2015.

[19] T. Watanabe, S. Ito and K. Yokoi, Co-occurrence histograms of oriented gradients for human detection, In Information and Media Technologies, Vol. 5, no. 2, pp. 659-667, 2010.

[20] U. Marti and H. Bunke, A full English sentence database for off-line handwriting recognition, Proceedings of ICDAR, pp. 705–708, (1999).

[21] V. Margner, N. Ellouze, H. Amiri, M. Pechwitz and S. Snoussi Maddouri, IFN/ENIT - database of handwritten Arabic words, Proceedings of CIFED, pp. 129–136, (2002).

[22] A. Mezghani, S. Kanoun, M. Khemakhem and H. El Abed, A Database for Arabic Handwritten Text Image Recognition and Writer Identification, Proceedings of ICFHR, pp. 399–402, (2012).

[23] F. Slimane, R. Ingold, S. Kanoun, A. Alimi and J. Hennebert, A New Arabic Printed Text Image Database and Evaluation Protocols, Proceedings of ICDAR, pp. 946–950, (2009).

[24] L. Ladha and T. Deepa, Feature Selection Methods and Algorithms, In International Journal on Computer Science and Engineering, Vol. 3 No. 5, pp. 1787-1797, 2011.

[25] D. A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, In 1st edition. Prentice Hall, 2002.

Dr. Afef Kacem Echi received M.Sc. and Ph.D. degrees in Computer Sciences from the National School of Computer Sciences of Tunis in 1997 and 2001 respectively. Since 2000, she has been an assistant in the computer science department at the Faculty of sciences of Monastir, and was appointed Assistant Professor there in 2002. Dr. Kacem is a responsible member of the research area: Analysis and Recognition of Handwriting and document in LaTICE: Laboratory for Technologies of Information and Communication at the National High School of Engineers of Tunis. She has authored over 50 articles in various national and international journals and conference proceedings.

Asma Saïdani received Master degree in Computer Sciences, in 2009, from the High School of Sciences and Techniques of Tunis. She is a member of research teams of LaTICE: Laboratory for Technologies of Information and Communication at the High School of Sciences and Techniques of Tunis.