A System for an automatic reading of student information sheets

Afef Kacem, Asma Saïdani, Abdel Belaïd

A System for an automatic reading of student information sheets

Afef KACEM, Asma SAÏDANI
UTIC, Higher School of Sciences and Techniques of Tunis
Tunisia
afef.kacem@esstt.rnu.tn, saidaniasma@yahoo.fr

Abdel BELAÏD
LORIA, Scientific campus
Vandoeuvre-Lès-Nancy
France
abelaid@loria.fr

***Abstract*—** In this paper we present a student information sheet reading system. Relevant algorithm is proposed to locate and label handwritten answer field. As information sheets can be filled in Arabic and/or in French, automating the script language differentiation is a pre-recognition required in the proposed system. We have developed a robust and fast field classification and script language identification method, based on a decision tree, to make these processing practical for sheet recognition. To this end, the system uses several novel features (loops, descenders, diacritics) and analyses the lower profile of script. The classification rates are 92.5% for numeric fields, 94.34% for Arabic scripts and 94.66% for French scripts. Experimental results, carried on 80 sheets, show our system provides an effective way to convert printed sheets into computerized format or collect information for database from printed sheets.

Keywords-Form segmentation; feature extraction; writing language identification;handwritten recognition

I. INTRODUCTION

Every day, millions of forms including applications, family allowances, subscription newsletters, inquiry about products, etc. have to be processed. A great deal of time, effort and money will be saved if it can be executed automatically. However, in spite of major advances in computer technology, the degree of automation in acquiring data from such documents is very limited and a great deal of manual labor is still needed in this area. Thus, any method which can speed up this process will make a significant contribution. This paper deals with the essential concepts of form analysis and recognition. It specially concerns the acquirement sheets needed for student subscription in our school.

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 student. The fixed fields are intended to identify the school, to inform the student or to question him. The fields are grouped into blocks. Blocks are clearly separated by white space or separators. With the proposed system, the student information sheets can be automatically analyzed and data captured from sheet fields. Thus, it will be possible to quickly validate poor quality sheets and then can be utilized for other tasks.

The outline of the paper is as follows. Section II exposes some related works. Then section III presents the actual state of the proposed system. The first principles of the system are described in subsections A and B in which we mainly focus on answer field extraction and classification and the script language identification. Section IV provides and discusses some obtained results. Finally, an analysis of the errors leads to the conclusion in section V.

II. STATE OF ART

Machine recognition of forms has been the subject of extensive research in the last decade. One of the principal applications of form analysis is the extraction and subsequent recognition of user entered information. Extraction of user entered information is an important preprocessing step to facilitate subsequent high accuracy recognition. As reported by [1], Gillies et al. proposed a census form processing system, Govindaraju et al. studied a system for handwritten document that included checks, Cesarini et al. proposed a system for data extraction from forms. However their techniques require that the class is known a priori. Also, Wang et al. studied form images characterized by simple background such as ruled lines and boxes. Liang et al. proposed a method to extract printed text strings from periodic background images but it is not suitable for extraction of handwritten characters or where the background is non-periodic [1].

As forms have begun to include different languages, automating the differentiation of writing has become a pre-recognition required in any system of automatic processing of multilingual documents. Three approaches can be used to design these systems: (1) global approach which handles the text blocks in their entirety, using features such as Gabor filters, Co-occurrence matrix, wavelets [2-6], (2) local approach which treats block text details based on analysis of text-lines, words or connected components using horizontal projection histogram or contour profile analysis or water reservoir [7-10] and (3) hybrid approach which combines the first two approaches using information on blocks, lines or words and connected components [11].

III. SYSTEM OVERVIEW

The system relies on a four steps process: 1) handwritten answer field extraction, 2) field classification and script language identification 3) handwritten recognition and 4)

collection of captured data for database. In this paper, we mainly focus on the first two steps.

A. Handwritten answer field extraction

The student information sheet has a completely static structure allowing immediate location of areas of interest. From these areas of interest delimiters, materializing the area in which the student has to write, easily identify the components belonging to the field looking for: predefined boxes, in case of pre-box area with the region containing information, baseline upon which the student must fill the field (see Figure. 1).

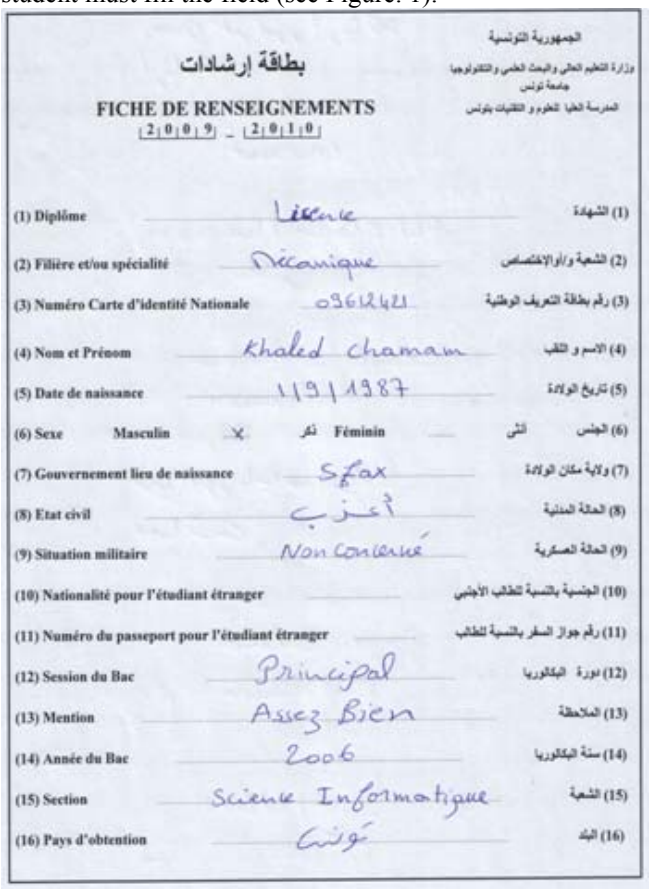


Figure 1. The front studend information sheet.

Here field libels are printed in Arabic and placed in the right side of the sheet. Just in front, in the left side of the sheet, we find their translation in French. Between each pairs of libels, the student can answer in Arabic of in French. To extract the text answer fields, the sheet is firstly segmented into lines by grouping linearly arranged connected components. Note that line extraction step also serves to label answer fields. We manually assign to each extracted line a label (diploma, specialty, First and last names, date birth, zip code, etc.) for database building and indexing. In case of overlapping lines when writing overflows, the system separates between lines by affecting the overlapped components to lines to which they are closes (comparing their center gravities to line ordinates, see figure 2). Secondlay the system inspects components along lines, from their extremities, and stops when it meets components different from points.

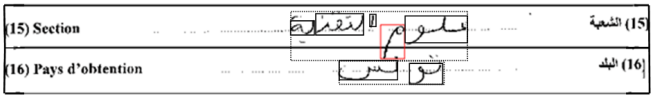


Figure 2. Examples of text answer fields.

To not discard diacritics that can be confused with the dotted line, the system only removes points in the band's central of text-lines (See Figure 3). We used mathematical morphology to enhance result of answer field extraction step such as image dilation for broken charcters.

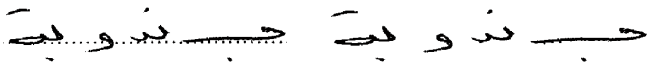


Figure 3. Dotted line remouving

To locate fields for ZIP code (see Figure 4) which is written in pre-box area, the system look for connected components that matchs in width with the predefined box template. When the corresponding fields found, the system extracts the field contents. Knowing that ZIP code would contain only four digits, the systme just need to extract four connected components which overlp with the pre-box area bounding box. This information could be also used to separate between digits which sticks with each other or with the pre-box area.

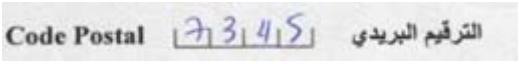


Figure 4. Pre-box area example

B. Answer field classification

This step allows the identification of digital fields and the script language for the remaining text answer fields. Languages taken into account by the prototype are French and Arabic. To classify the answer field content, we must first extract some features out the field area. We will use these features inside a classification tool, here a decision tree, to obtain the final class.

Digital fields are generally small in number of connected components. These components are often aligned, close to each other and regularly separated (see Figure 5).

Figure 5. Numeric answer field

To identify the writing language, we used some intuitive characteristics of handwritten scripts (loops, diacritics,

descenders, profiles) to distinguish between Arabic and French scripts. Let us quote some of them. In Arabic, the loops are generally found in the central band (ة, ص,ض, ط, ظ, ع, ق, ص,م, ف, ق, م, و, etc.), except for the letter (ه) in which the loops are lightly beyond the central band. In Latin, letters that have loops in the central band are : uppercase *B* and lowercase *a, b, d, g, k, o* and *q*. Letters, containing loops over the central band are uppercase *A, B, P, R* and lowercase *f* and *l*. Letters containing loopsbelow the central band are lowercase *j, f, g, y* and *z*. To benefit from this ascertainment, we extract then classify loops relative to the central band (See Figure 6).

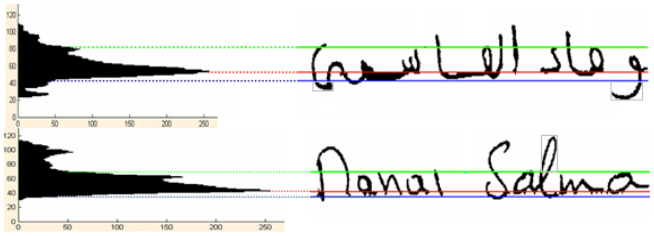


Figure 6. Arabic/French identification based on loop positions.

We also find that some arabic letters (ص, ق, خ, ح, ش, س, ر, غ, ع, ز, و) has descenders which are horizontally extended (having un aspect ratio, widh/height greater than one). Descenders in French scripts are rather vertically extended (See Figure 7).

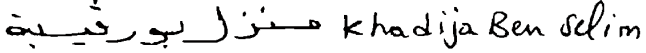


Figure 7. Descenders in Arabic/French scripts.

Moreover, Arabic script uses more diacritical points (0 to 3 per letter without considering others diacritic signs such as chedda, hamza, soukoun, fatha, kasra, dhamma used in vowel Arabic) compared to French script. Indeed, many are Arabic letters that have the same body but not the same number and location of diacritics, for example letters ب, ت , ث. These diacritic points can be below or above the central band of Arabic word. In French, only the letters *i* and *j* which have points over their body (see Figure 7). Furthermore, no letter in French script has diacritic points below the central band.

Comparing the lower profile of French and Arabic scripts, we note that Arabic script is generally flattened compared to French script (see Figure 8).

Figure 8. Lower profile of connected compnents.

To compute the lower contour profile of script, we firstly must have a sufficient amount of information to characterize the texture of the answer fields, and then we have to normalize these fields according the maximum field size because these fields do not have the same size and the same content. Next, we must duplicate the field content respecting a regular space between words or parts of words without exceeding the maximal size. We finally get, for each component column, the lowest black pixel. Thus, for a component of width N, its lower profile should contain N pixels (see Figure 9).

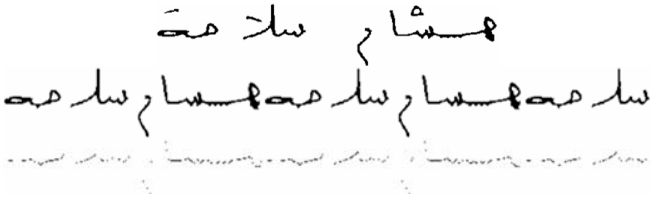


Figure 9. Field normalisation and Lower profile.

The continuity of the lower profile is computed along differences between pairs of sucessif pixels p_i and p_{i+1} as follows, y_{pi} is the i^{th} pixel ordinate:

$$d_i = \left| y_{p_{i+1}} - y_{p_i} \right|, \quad 0 \le i \le N-1$$

For a connected component, the total distance of its lower contour is calculated as follows:

$$bd(j) = \sum_{i=0}^{N-1} d_i$$

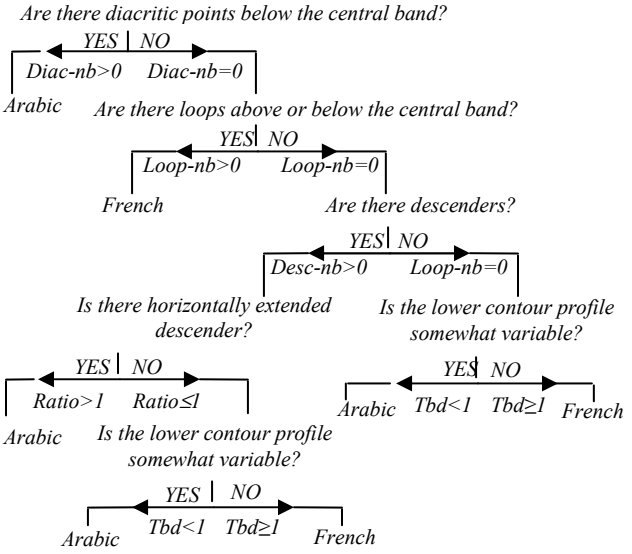
For a field, consisting of M connected components, the global distance of the lower contour is calculated as follows:

$$tbd = \sum_{j=1}^{M} bd(j)$$

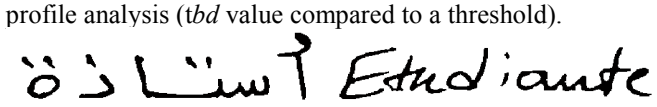
Notice that *tbd* of Arabic script is generally inferior than French script one, because Arabic writing is more straight and flattened and has no high connections between characters like what we find in French writing especially when we used letters such as *o, v*. High links between characters increases differences between pixel ordinates and so the *tbd* value.

After normalizing *tbd* values, we set a threshold equal to 1, which differentiates between Arabic and French scripts. Thus, if *tbd* is less than 1, the field script language is classified Arabic, otherwise it is classified French. In Figure 9, the *tbd* of the text, written in Arabic, is equal to 0.67.

Now, to identify the script language, we used the overall structural features, previously extracted, (number of diacritical signs below the central band, number of loops above and below the central band, number and ratio aspect of descenders) and *tbd* value which inform about script continuity. The algorithm, proposed to identify the script language, is displayed as decision tree which (1) is simple to understand and interpret, (2) uses a white box model (if a given result is provided by a model, the explanation for the result is easily replicated by simple math and 3) can be combined with other decision techniques.



Figures 10 presents two words which do not have diacritical points below the central band, neither descenders nor loops above or below the central and whose script language was correctly identified thanks to the their lower profile analysis (t*bd* value compared to a threshold).



(a) tbd=0.84<1 *(b) tbd=1.25>1*

Figure 10. Arabic and French script *Tbd*

C. *Handwritten recognition*

It involves the automatic conversion of the handwritten text answer field area into letter codes which are usable within computer and database applications. Handwriting recognition is difficult, as many students have different handwriting styles. And, as of today, OCR engines are primarily focused on machine printed text and ICR for handwritten text. There is no OCR/ICR engine that supports handwriting recognition as of today.

Several different recognition techniques are currently available. Techniques ranging from statistical methods to machine learning like neural networks or support vector machines have been applied to solve this problem. But since handwriting depends much on the writer and because we do not always in exactly the same way, building a general recognition system that would recognize any script with good reliability in every application is not possible.

Typically, the recognition systems are tailored to specific applications to achieve better performances. In particular, unconstrained handwritten digit recognition can be applied here to recognize ZIP codes, phone number, birth date, graduation date, etc.

Moreover, narrowing the problem domain often helps increase the accuracy of handwriting recognition step. A field for a ZIP code for example, would contain only the characters 0-9. This fact would reduce the number of possible identifications. Furthermore, the recognition of 4-digit ZIP code part here can be cross-validated with those of city and state names recognition.

We tried some available tools however there are several common imperfections in this step. The most common being characters that are connected together are returned as a single sub-image containing both characters (See Figure 11). This causes a major problem in the recognition stage.

Figure 11. Connected characters

IV. EXPERIMENTS

We carried out recognition experiments on a variety of forms (about 80 sheets) scanned at a resolution of 600 dpi and stored in the format bitmap. Table I displays results provided by the handwritten answer field extraction step.

TABLE I. REULTS OF ANSWER FIELDS EXTRACTION

Answer fields	Good extraction rate	Failure rate
900	95.88 %	4.12%

Some extraction errors appear when the student does not respect the area limits fixed for answer fields (see Figure 12). To be able to correctly extract the handwritten text answer field, the system should distinguish between printed and handwritten scripts to solve such problem.

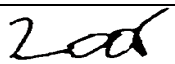
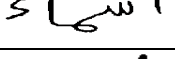
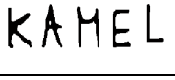
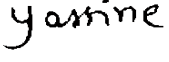
Figure 12. Problem in answer field content extraction

To evaluate the script language identification step, the system is tested on 800 answer fields: 300 written in Arabic, 300 written in French and 200 numeric fields. Table II gives an overview of the obtained results.

TABLE II. RESULTS OF SCRIPT LANGUAGE IDENTIFICATION

Script	Identification Rate	Confusion rate	Confusion Matrix		
			Numeric	*Arabic*	*French*
Numeric	92.50%	7.5%	92.50%	0%	7.50%
Arabic	94.34%	5.66%	0%	94.34%	5.66%
French	94.66%	5.34%	3.34%	2%	94.66%
Mean	93.83%	6.17%	-	-	-

Observing some confusion cases, we find that most of errors are due to handwriting variability as explained in Table III.

Script	Actually type	Output	Confusion origin
	Numeric	French	*tbd*=1.2>1
	Numeric	French	Loop above the central band
	Arabic	French	Loop below the central band
	Arabic	French	*tbd*=1.41>1
	French	Numeric	Morphological similarity
	French	Arabic	Horizontally extended descender

V. CONCLUSION

Form analysis and recognition is widely known problem. As previously mentioned, many related works have been proposed. There exists nowadays, software which are able to analysis forms, but they remain out of reach because of their high cost. In addition, these programs are not easily adaptable to different types of forms. As consequence, we decided to develop our own solution. The proposed system aims to recognize student information sheets filled in by hand in order to automate data entry tasks. It starts by locating answer field areas. From which, it extracts some structural and global features (diacritics, loops, descenders, lower contour profiles) and uses them through a decision tree to classify the answer field contents. The decision tree seems to be a helpful tool to distinguish between Arabic and French scripts. It provides a highly effective structure within which the system has been able to lay out options and investigate the possible outcomes of choosing those options.

Experimental results shows that the proposed feature extraction and field classification methods are accurate and easy for extension.Our system achieved an average rate of script language identification of 93.83% which indicates that the proposed approach is very suitable for information sheet recognition. Moreover, an analysis of the errors is conducted to discuss possible means of enhancement and their limitations.

As future work, we plan to develop tests of efficiency and robustness of our system on wider database of sheets. Also, the decision tree which was manually built (based on the features we suggested to handle a specific descrimination task), could have probably be trained using any general machine learning algorithm. By doing so, the system could have been designed to be a much more generic. We also aim to improve the accuracy of field extraction and script language identification through the use of additional features such as Gabor filters. These features will be the topic for a future paper.

REFERENCES

[1] M. Okade and M. Shridhar, "Extraction of user entred components from a persoal bank check using morphological substraction", Automatic bankcheck processing (1997) 1-4, Word scientific publishing company,

[2] G. S. Peake and T.N. Tan, "Script and Language Identification from Document Images", Proc. of the Workshop on Document Image Analysis, 1997, pp.10-17.

[3] V. Singhal, N. Navin and D. Ghosh, "Script-based classification of Hand-written Text Document in a Multilingual Environment", Research Issues in Data Engineering, 2003, pp. 47.

[4] S. L. Wood, Xiaozhong Yao, K. Krishnamurthi and L. Dang, "Language identification for printed text independent of Segmentation", Proc. of ICDAR'95, vol.3, 1995, pp.3428-3431.

[5] Ben jlail M., Kanoun S., Alimi A.M.and Mullot R., " Three decision levels strategy for Arabic and latin textsdifferentiation in printed and handwritten natures", Proc. of ICDAR'07, vol. 2, 2007, pp. 1103-1107.

[6] Baâti K, Kanoun S and Benjlaiel M.,"Différenciation d'écriture arabe et latine de natures imprimée et Manuscrite par approche globale", Proc.of CIFED'2010.

[7] S. W. Lee and J. S. Kim,"Multi-lingual, multi-font and multi-size large-set character recognition using self-organizing neural network", Proc. of ICDAR'95, Vol.1, 1995, pp.28-33.

[8] A. M. Elgammal and M. A. Ismail, "Techniques for Language Identification for Hybrid Arabic-English Document Images", IEEE Proc. of ICDAR'01, 2001, pp.1100-1104.

[9] U. Pal and. B. Chaudhuri,"automatic Identification of English, Chines, Arabic, Devangari And Bangla Script Line", Proc. of ICDAR'01, 2001, pp.0790-0794.

[10] L. Zhou, Y. Lu and C. Lim Tan, "Bangla/English Scrip Identification Based on Analysis of Connected Component Profiles", Proc. 7th DAS, 2006, pp. 243-254.

[11] U. Pal and B. Chaudhuri,"Script Line Separation from Indian Multi-Script Documents", Proc. ICDAR'95, pp.406-409, 1999.