Electronic Letters on Computer Vision and Image Analysis 13(1):1-17, 2014

How to separate between Machine-Printed/Handwritten and Arabic/Latin Words ?

Afef Kacem Echi[1], Asma Saïdani[1] and Abdel Belaïd[2]

[1] *University of Tunis, ESSTT-LaTICE, 5 Avenue Taha Husseïn, BP 56 Babmnara 1008 Tunis, Tunisia*

[2] *University of Lorraine, LORIA, Campus scientifique. BP 239 54506 Vandoeuvre-lès-Nancy, France*

Received 11 December 2013; Accepted 11 April 2014

Abstract

This paper gathers some contributions to script and its nature identification. Different sets of features have been employed successfully for discriminating between handwritten and machine-printed Arabic and Latin scripts. They include some well established features, previously used in the literature, and new structural features which are intrinsic to Arabic and Latin scripts. The performance of such features is studied towards this paper. We also compared the performance of five classifiers: Bayes (AODEsr), *k*-Nearest Neighbor (*k*-NN), Decision Tree (J48), Support Vector Machine (SVM) and Multilayer perceptron (MLP) used to identify the script at word level. These classifiers have been chosen enough different to test the feature contributions. Experiments have been conducted with handwritten and machine-printed words, covering a wide range of fonts. Experimental results show the capability of the proposed features to capture differences between scripts and the effectiveness of the three classifiers. An average identification precision and recall rates of 98.72% was achieved, using a set of 58 features and AODEsr classifier, which is slightly better than those reported in similar works.

Key Words: Script and nature Classification, Feature extraction.

1 Introduction

There are many document analysis systems which are able to handle single language documents. But there is a big need to expand these systems to handle multi-lingual documents. To develop such systems, the problem of script identification has to be addressed. The term script identification refers to the task of identifying the language a given document is written in. It plays a major role in several applications. It is mostly used as an important preprocessing step in the design of an OCR (Optical Character Recognition) system. In past years, some works have been done on script identification. They mainly depend on various features extracted from document images at text-block [3], [7], [11], [27], [29] text-line [3], [8], [10], [28] or word level [4], [5], [12], [13]. As mentioned by [14], block level script identification identifies the script of the given document in a block and concerns documents written in different languages. In text-line based script identification, a document image can contain more than one script and the script is written occasionally to highlight one sentence.

Correspondence to: <afef.kacem@esstt.rnu.tn>

Recommended for acceptance by <Umapada Pal>
ELCVIA ISSN:1577-5097
Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

Word level script identification allows the document to contain more than one script and the words are scattered throughout the document whenever it is necessary. Notice that most of the existing systems work on block level script identification. As they are based on the overall visual appearance of the text-block, they are generally incapable of tackling the variations in the writing style, character style and size, spacing between lines or words, etc. When the classification is performed by words and not by text-line or text-block, it will be possible to analyze more cases with scripts more or less long, written in the form of words or lines. But this requires finest analysis of each word. In this work, many attempts have been made to identify the script (Arabic or Latin) and its nature (printed-machine or handwritten), at the word level. Various features, extracted from word image are presented, tested and evaluated under the same experimental conditions. The objective is to contribute to the field of script and nature identification through better selection and combination of features, used in the literature, with those here proposed. This paper aims providing the reader with multiple points of reference useful for comparing a number of published results and a proposal set of features that appears to be interesting for Arabic/Latin and handwritten/machine-printed scripts identification. The remainder of the paper is organized as follows: Sect. 2 gives a brief description of the properties of Arabic and Latin scripts. Afterwards, Sect. 3 provides an overview about Arabic/Latin and handwritten/machine-printed scripts identification approaches. Sect. 4 emphasizes on feature extraction and selection. Sect. 5 discusses experimental results. Finally, Sect. 6 summarizes the conclusions and future work.

2 Characteristics of Arabic and Latin Scripts

We present the aspects that Arabic has in common with Latin as well as those different.

2.1 Similarities of Arabic and Latin scripts

Arabic and Latin writings have in common the following points:

- **Presence of writing lines**: (See Fig. 1):

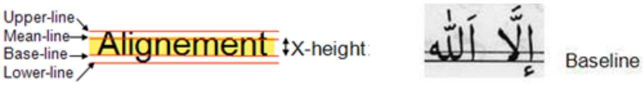


Figure 1: Writing Lines.

- **Central band**: It is generally the most loaded in terms of information density in pixels (See Fig. 2). It corresponds to the places of the horizontal ligatures and to the centered letters (without extensions).

دار الكتب الوطنية
Bien

Figure 2: Central band computed for text-line horizontal projection.

- **Regularities and singularities**: The Arabic and Latin scripts presents the singularities upper the central band and the regularities inside the central band (See Fig. 3).

- **Inter-writer variability**: Different people have different style of writing which results in inconsistent shapes for the same letters (See Fig. 4). Also, random variations in shapes are encountered across different instances of letters written by the same scribe.

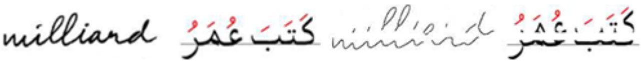


Figure 3: Singularities on either side of the central band and regularities inside.

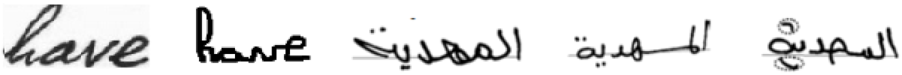


Figure 4: The same word written by different writers.

2.2 Differences between Arabic and Latin Scripts

Arabic is different from Latin with respect to a number of aspects:

- **Writing orientation**: It is right to left for Arabic and from left to right for Latin.
- **Horizontal projection**: As illustrated in Fig. 2, the horizontal projection profiles of Arabic texts have a single peak around the middle of the text-line. In contrast, projections of Latin texts have two major peaks.
- **Alphabet**: The Arabic alphabet consists of 28 letters without considering the variation of their shapes according to the position, the voyellation elements and the phonetic context.
- **Word**: Arabic letters are strung together to form words in one way only. There is no distinction between printing and cursive, as there is the case in Latin.
- **Letter shape**: Neither are there capital and lowercase letters, all the Arabic letters are the same. The letter shapes, however, are changeable in form, depending on the location of the letter at the beginning, middle or at the end of the word. Some connect only on one side, others on both (See Fig. 5).

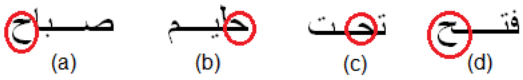


Figure 5: Different shapes of a same Arabic letter.

- **Diacritic**: Arabic writing is very rich in diacritic marks (e.g. dots, hamza, etc.). Some Arabic letters may have exactly the same main shape, and are distinguished from each other only by the presence or the absence of these diacritic, their number and their position with respect to the main shape. Diacritic points can be located above or below the letter, but never both simultaneously. Arabic letters in Fig. 6 are five similar letters with one, two or three diacritical points above or below the letter body. A Latin text is lower in diacritic compared to an Arabic text. There are only the two letters 'i' and 'j' which have only one diacritic point above. There are no diacritic at the bottom in a Latin text.
- **Semi cursive writing**: Arabic writing, both handwritten and printed, is semi-cursive: the word is a sequence of connected components called PAWs (Piece of Arabic Word). Each PAW is a sequence of completely cursive letters. PAWs are separated by small blanks and not necessarily composed of the same number of letters (See Fig. 7).
- **Ligatures**: Letters of a PAW could be horizontally or vertically tied (in some fonts, two, three or even four letters can be tied vertically) as shown in Fig. 8.

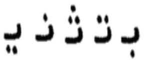


Figure 6: Different diacritic point number for the same main shape of Arabic letter.

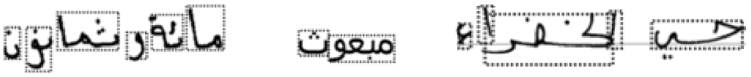


Figure 7: Semi-cursive writing.

- **Elongations**: The same Arabic word does not have a fixed length since different elongations numbers could appear between letters (See Fig. 9):

The foregoing features relate to the Arabic writing whether printed or handwritten. In case of manuscript, others specificities are involved:

- **Fusion of diacritical points**: Two or three diacritical points can easily be agglomerated in two or even one diacritical point (See dotted circles in Fig. 4).
- **PAW Overlapping**: In one word, two consecutive PAWs may overlap (See last word in Fig. 7).

3 Literature Review

The surveyed methods are summarized in Table 1. They mainly concern Arabic/Latin and handwritten/machine-printed scripts identification.

Table 1: Identification Method Summarization

Ref.	Script	Nature	Level	Identification Rate
[3]	Arabic/Latin	Printed/Handwritten(400 Blocks)	Block/Text-line/Connected component	88.5%
[4]	Farsi_Arabic	Printed/Handwritten (32006 Words)	Word	97.1%
[5]	Latin	Printed/Handwritten (Public databases)	Word	Above 80%
[6]	Bangla/English	Handwritten (1200 Blocks)	Connected component	95%
[7]	Arabic/Latin	Printed/Handwritten (400 Blocks)	Block	95%
[8]	Latin	Printed/Handwritten (50 Documents)	Text-line	98.2%
[10]	Arabic/English	Printed (1976 Text-lines, 8320 Words)	Text-line/Word	99.7%(Text-line), 96.8%(Word)
[11]	Arabic/Latin	Printed/Handwritten (800 Documents)	Block	84.75%
[12]	Arabic/Latin	Printed/Handwritten (800 Words)	Word	97.5%
[13]	Arabic/Latin	Printed(learning: 3383 Words, test: 846 Words)	Word	94.32%
[24]	Arabic/Latin	Printed/Handwritten (learning: 400 Documents, test:200 Documents)	Document	82%(Arabic), 92%(Latin)

4 Feature Extraction

This section summarizes and organizes the information available in the literature in an attempt to motivate researchers to look into the proposed features and try to develop more advanced ones. Several feature sets, used in the literature or proposed here, are used to illustrate their properties and performances.

Figure 8: Vertical and horizontal ligatures in Arabic.

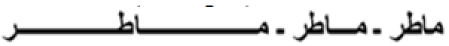


Figure 9: Writing elongations.

4.1 Features Proposed in the Literature

- **Vertical projection variance of the word**: Due to overlaps between handwritten words, projection profiles has smoother valleys and peaks resulting in smaller variance compared to machine-printed words [11]. The variance of the vertical co-ordinates of the vertical projection profile is calculated as a measure of homogeneity of the projection profile. In our point of view, this feature can be used to separate handwritten from machine-printed words either they are written in Arabic or Latin (See Fig. 10).

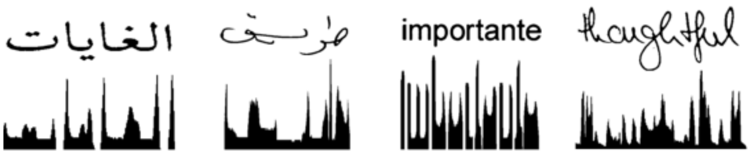


Figure 10: Vertical projection variance of the word.

- **Connected component width, height, aspect ratio, area and density**: As underlined by [7], the sizes of connected components inside a machine-printed word are more consistent, leading to smaller width and height variances. Each of these features is generalized in terms of mean and standard deviation. According to us, these features can only discriminate between handwritten and machine-printed Latin words but not between handwritten and machine-printed Arabic words. In fact, in Arabic there is no distinction between printing and cursive. Moreover, the same Arabic word does not have a fixed length on account of the elongations.
- **Separator length between two successive connected components**: In printed Latin, connected components are separated by regular separators as noted by [7] and shown in Fig. 11. This feature is generalized in term of mean and standard deviation. In our opinion, this feature can just identify the nature (handwritten/machine-printed) of Latin words but not of Arabic words because of PAW overlapping.
- **Connected component profiles analysis**: Handwritten and machine-printed Arabic is cursive. It is also the case of handwritten Latin but not for machine-printed Latin. As done in [6], we extracted the bottommost profile of the connected components (after elimination of diacritic points) i.e. the lowest pixels of vertical columns of the components. To obtain the bottommost profile, each vertical column of a particular connected component is scanned from bottom until it reaches a black pixel P_i. Thus, for a component of width N, we get N such pixels. For examples of bottommost profiles, see Fig. 12. To measure the discontinuity of bottommost contour line of the component, we traverse from P_i to P_{i+1} and obtain the difference d_i of two adjacent pixels of the components, which is computed as follows where y_{p_i} is the coordinate value of the pixel P_i.

$$d_i = |y_{p_{i+1}} - y_{p_i}|, 0 \leq i \leq 1 \tag{1}$$

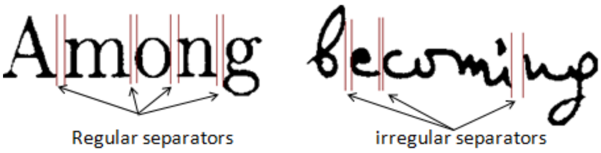


Figure 11: Separator length between two successive connected components.

The total distance of the bottom border of the component is computed as:

$$bd(j) = \sum_{i=0}^{N} d_i \tag{2}$$

The aggregate distance of the bottom pixels is then computed as follows where M is the number of the word connected components and W is the word width:

$$tbd(j) = \frac{\sum_j^M bd(j)}{W} \tag{3}$$

As noted in a previous work [1], [2], Arabic script has lower *tbd* than Latin because Arabic script has lower discontinuity since it is straighter and flat and does not have high links between letters like in handwritten Latin script (especially in case of *o*, and *v* letters). In fact, the high links increase the differences between coordinates of lower profile pixels of word. Fig. 12 illustrates some discriminative examples based on connected component profiles analysis.

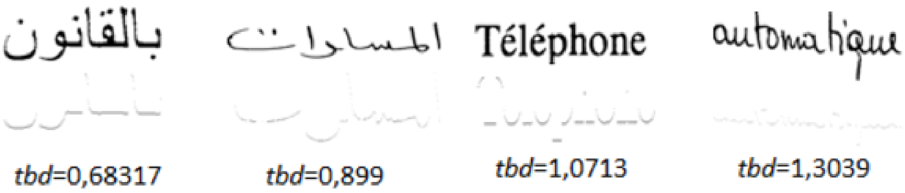


Figure 12: Connected component profiles analysis.

- **Loop aspect ratio**: Printed script is a succession of connected components comprising loops whose surfaces are regulars contrary to the handwritten Arabic and Latin which depend on the writing style [3] (See Fig. 13). The loop aspect ratio is considered as feature and generalized in terms of mean and standard deviation.

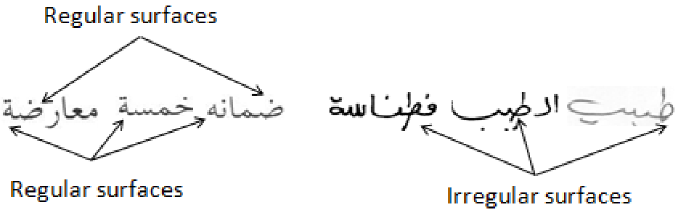


Figure 13: Loop aspect ratio.

- **Pixels distribution**: This feature is especially used to discriminate handwritten from printed Latin in [8]. The bounding box is divided in two by a horizontal line as indicated by the red line (See Fig. 14). The bounding box height is decreased by 10 pixels as shown by the green lines. 14. Then the density of the upper part and of the lower part is calculated. We retain the difference between these densities as feature.

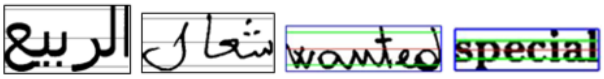


Figure 14: Bounding box horizontal division.

- **Baseline profile features**: Most pixels of machine-printed words are located on the baseline. In the machine-printed words, position of ascender and descender is determined by the baseline. However, in handwritten case words are not usually written in a single baseline and position of ascender and descender varies according to the writer's style. Difference between handwritten and machine-printed words is shown in the baseline profile features [4]. The baseline is estimated as the peak of horizontal histogram of the word image (See Fig. 15). The following features are extracted from the baseline profile: baseline position, sub-baseline number n (a sub-baseline represents pixels of the word image that lays on the baseline as shown in. 16), distance of highest scan line from the baseline (d_1), distance of lowest scan line from the baseline (d_2) and the number of pixels on the baseline (p). Mean, variance of sub-baselines,

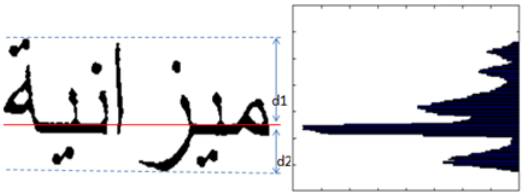


Figure 15: Baseline profile features.

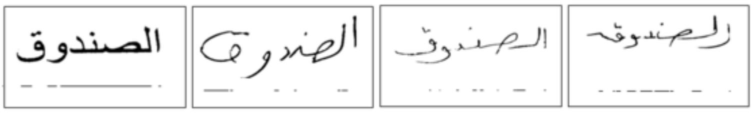


Figure 16: Sub-lines of an Arabic word.

and ratio of sub-baseline to their variances are also taken as features (See Table 2).

- **Run-length histogram**: [9] proposed features from Run-length histogram for machine-printed/handwritten Chinese character classification. We think that these features can be used to underline the difference between the stroke length of machine-printed and handwritten words. We extracted black pixel run-lengths in three directions, including horizontal, vertical and diagonal. We then calculated three histograms of run-lengths for these directions. To get scale-invariant features, we normalized the histograms. The normalized histogram C'_k is calculated according the following equation where C_k is the number of runs with length k and N is the maximal length of all possible runs.

$$C'_k = \frac{C_k}{\sum_{i=1}^{N} C_i} \tag{4}$$

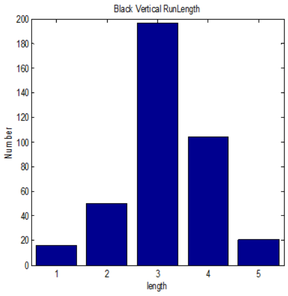
Table 2: Baseline profile features.

Word	d_1	d_2	n	p	Mean	Variance	Ratio
الأرض	129.97	170.26	5	054	38.20	42.75	0.117
ربا بيع	86.98	91.05	3	0.34	29.66	41.00	0.073
fonctions	65.98	72.09	11	0.73	53.81	9.09	1.20
Premier	55.98	72.23	12	0.50	10.83	4.40	2.72

To get the final features, the histogram is then divided into five bins with equal width and five rectangular-shaped weight windows are used. Thus, we extracted five features in each direction, leading to 15 features. Table 3 presents the black vertical run-length of some words. We noted that run length values are high for handwritten words.

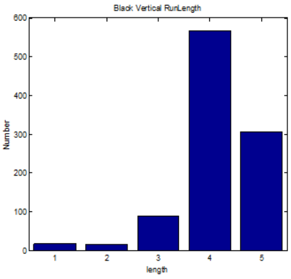
Table 3: Black Vertical Run length histogram.

machine-printed Arabic



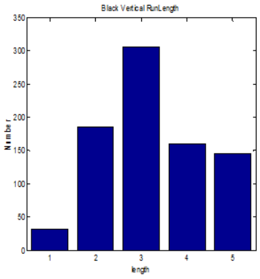
ميزانية

handwritten Arabic



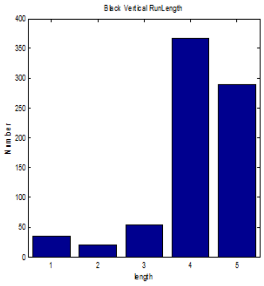
متكا فئ

machine-printed Latin



algorithme

handwritten Latin



simples

- **Crossing count histogram**: Crossing count is the number of transitions from 0 to 1 along a hypothetical horizontal or vertical line over the word image. Crossing count features are already used by [4] for Latin handwritten and machine-printed Farsi_Arabic words discrimination. These features are used to measure stroke complexity. For each horizontal and vertical scan line, the crossing count is calculated. Horizontal and vertical crossing counts are defined as follows where $I(x, y)$ designates the pixel at the position (x, y) of the image I.

$$Profile_{cchorizontal}(y) = \sum_{x} (1 - I(x, y)) * I(x + 1, y) \tag{5}$$

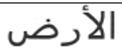
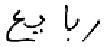
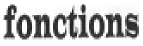
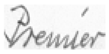
$$Profile_{ccvertical}(x) = \sum_{x} (1 - I(x, y)) * I(x, y + 1) \tag{6}$$

We then get two histograms for the horizontal and vertical crossing counts respectively. To have the final features from the histograms, the same technique, used to extract the run-length features, is exploited.

- **Bi-level co-occurrence**: As defined in [15], a co-occurrence count is the number of times a given pair of pixels occurs at a fixed distance and orientation. For binary images, the possible co-occurrence pairs are white-white, black-white, white-black and black-black. As the black-black pairs carry most of the information than the other co-occurrence pairs, we only considered them to extract related features. We used horizontal H, vertical V, major MD and minor diagonal mD orientations and 2 pixels distance d level for the classification (See Table 4). The horizontal co-occurrence count $C_h(d)$ is defined as follows.

$$C_h(d) = \sum_x \sum_y I(x, y), I(x + d, y) \tag{7}$$

Table 4: Bi-level Co-occurrence

Word	H	V	MD	mD
	0.2526	0.2552	0.2454	0.2468
	0.2593	0.2544	0.2338	0.2525
	0.2555	0.2505	0.2472	0.2468
	0.2778	0.2624	0.2128	0.2470

- **Upper_lower profile**: As noted in [8], where authors tried to discriminate machine-printed from handwritten Latin text, using simple structural characteristics, the height of printed characters is more or less stable within text-line. On the other hand, the distribution of the height of handwritten characters is quite diverse. These remarks stand also for the height of the main body of the character as well as the height of both ascenders and descenders. Thus the ratio of ascender height to main body's height and the ratio of descender's height to main body's height would be stable in printed text and variable in handwriting. To characterize a word, based on its upper_lower profile, we extracted the following features: R_1 (the ratio of ascender zone to the main body zone), R_2 (the ratio of descender zone to the main body zone) and R_3 (the ratio of the area to the maximum value of the horizontal histogram of the upper-lower profile). Fig. 17 gives an example of these features computing on machine-printed Arabic word. Notice that connected components of diacritic points are not considered in the analysis of the upper_lower profile.

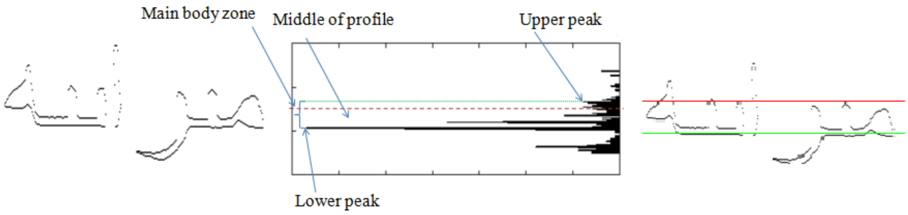


Figure 17: Upper_Lower profile.

- **Word physical sizes**: [4] noted that the sizes of machine-printed words are more consistent than those of handwriting on the same form. Thus, features related to the physical sizes of the word block such as density of black pixels, width, height, aspect ratio and area are considered.
- **Overlapping areas**: Unlike machine-printed words, for a handwritten word, the bounding boxes of the connected components tend to overlap with each other [4]. The overlapping area, normalized by the total area of the block is calculated as feature (See Fig. 18).
- **Moments**: Simple properties of the word image can be found via image moments. They include area, its centroïd and information about its orientation. So, we considered both central moments (centroid coordinates: X_g and Y_g) and *Hu* moments (only h_1 and h_2) as shown in Table 5.

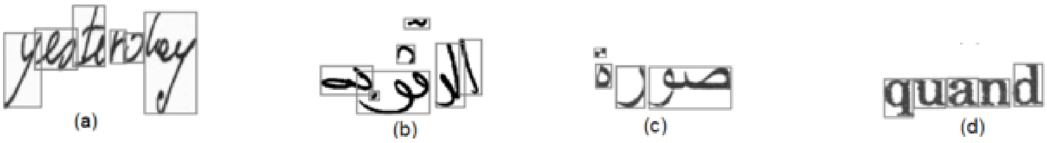
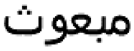
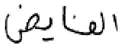
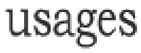
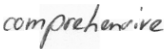


Figure 18: Overlapping areas: (a) and (b) handwritten words, (c) and (d) machine-printed words

Table 5: Examples of word image moments computing.

Word	Central moments (X_g, Y_g)	*Hu* Moments (h_1, h_2)
	(155.9144, 60.1865)	(0.3401, 0.0574)
	(158.2620, 42.1019)	(0.3755, 0.1025)
	(164.0405, 41.3795)	(0.5158, 0.2093)
	(152.1264, 52.7418)	(0.2955, 0.0532)

- **Steerable pyramid transform**: Steerable pyramid decomposition is a linear multi-orientation, multi-resolution image decomposition method, by which an image is subdivided into a collection of sub-bands localized at different scales and orientations (See Fig. 19). Features extracted from pyramid sub-bands served, in [12] to classify the scripts on only one script among the scripts to identify.

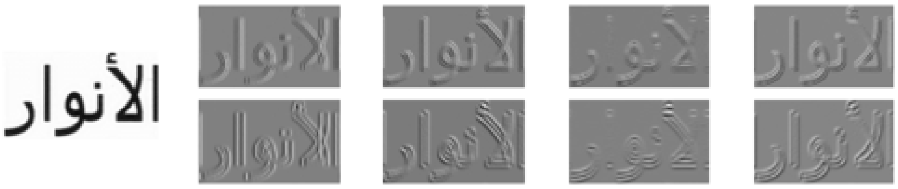


Figure 19: Decomposition with 2 levels and 4 orientations of a printed Arabic word

- **Gabor filters**: Gabor filter is a linear filter used for edge detection. Frequency and orientation representations of Gabor filters are similar to those of the human visual system, and they have been found to be particularly appropriate for texture representation and discrimination. In [16], Gabor filters are applied and 16 channels of features are extracted to identify the script (English or Chinese) of machine-printed words in scanned document images. In [24], authors differentiated Arabic and Latin texts using Gabor filters. Experimental results show the capability of Gabor filters to capture script features.

4.2 Features Proposed in this Work

We propose to extract some structural features distinctive to each type of writing. In fact, structural features are intuitive aspects of writing, such as loops, branch-points, end-points and dots. They mostly affect mostly the physical structure of words. Some structural features such as PAWs, ascenders, descenders, loops and upper and lower diacritic points considering their position in the word are already used in [13] to identify printed Arabic and Latin scripts. Here, we propose to test with some new structural features which include:

- **Presence of bottom diacritic points**: There are no diacritic points at the bottom in Latin words which is not the case of Arabic script (See Fig. 20).

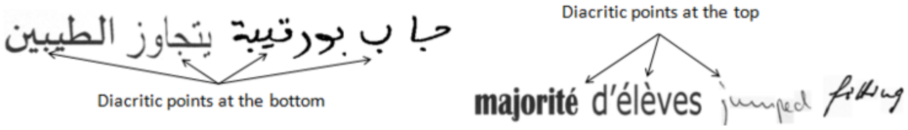

Diacritic points at the top
حا ب بورقيبة يتجاوز الطيبين
Diacritic points at the bottom
majorité d'élèves jumped fitting

Figure 20: Diacritic positions.

- **Loop position**: Loops in Arabic are generally written in the central band of the word with the exception of one Arabic letter in which the loops protrude slightly above and below. In Latin, there are a lot of letters which have loops above and below the central band (See Fig. 21).

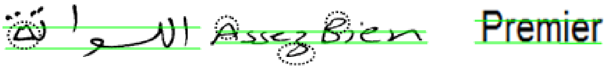


Figure 21: Loop position.

- **Presence of elongate descenders**: Arabic script, whether printed or written, is characterized by the frequently presence of elongated descenders. In Latin script, descenders tend to be vertical (See Fig. 22).

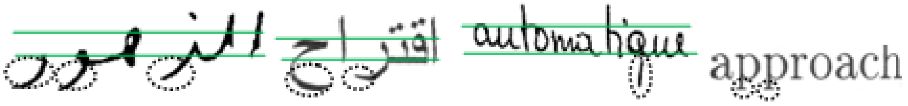


Figure 22: Descender shape.

Because some of studied or proposed features could be unnecessary or even redundant, their suitability should be analyzed. Feature selection aims to find the best subset of features that perform better than the original ones, and also, results in a more efficient classifier. Notice that feature selection algorithms can be broadly classified into wrappers and filters methods. Wrapper methods use a predictive model to score feature subsets. Each new subset is used to train a model, which is tested on a hold-out set. Counting the number of mistakes made on that hold-out set (the error rate of the model) gives the score for that subset. As wrapper methods train a new model for each subset, they are very computationally intensive, but usually provide the best performing feature set for that particular type of model. Filter methods use a proxy measure instead of the error rate to score a feature subset. This measure is chosen to be fast to compute, whilst still capturing the usefulness of the feature set. Filters are usually less computationally intensive than wrappers, but they produce a feature set which is not tuned to a specific type of predictive model. Many filters provide a feature ranking rather than an explicit best feature subset, and the cutoff point in the ranking is chosen via cross-validation. We tried various feature selection algorithms such as Principal Component Analysis (PCA) [17], Ranking [18], Genetic algorithm [19] and BestFirst [20].

5 Experimental Results Analysis

Experiments have been carried using two public databases: IAM database for Latin handwritten and IFN-ENIT for Arabic handwritten words. For Latin and Arabic machine-printed scripts, we created our own database by extracting words from various magazines and newspapers which contain variable font styles and sizes. A scanning resolution of 300dpi is employed for digitization of all the words (See Fig. 23). The training and test words have 1720 samples each, consisting of equal number of Printed Arabic (PA), Handwritten Arabic (HA), Printed Latin (PL) and handwritten Latin (HL) words. In the training phase, the features and the correct classification are used. For the test, we used the cross validation method: the words were divided into ten non-overlapping sets. Each time a classification model was calculated with training examples taken from nine sets and evaluated on the remaining sets. This procedure was repeated ten times. Each time using a different set as training examples. In Table 6, we give the correct identification rates for each proposed and previously used feature using Bayes (AODEsr classifier)[21]. The numbers between parentheses in the first column correspond to the number of measures representing the feature.

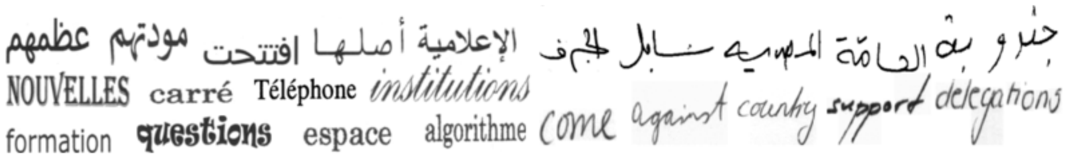

عظمهم موذتهم افتتحت أصلها الإعلامية
NOUVELLES carré Téléphone institutions
formation questions espace algorithme come against country support delegations

Figure 23: Some words used in the experiment

Note that we used AODEsr classifier which is a probabilistic classification learning technique based on the Bayes rule of conditional probability. Recall that Naive Bayes classifier works on a simple, but comparatively intuitive concept. It uses all the attributes contained in the data (features extracted from word image), and analyzes them individually as though they are equally important and independent of each other. In some cases, it is seen that Naive Bayes outperforms many other comparatively complex algorithms. In fact, we tested with many classifiers, as it will be shown later, and we found that AODEsr classifier which was developed to address the attribute-independence problem of the popular Bayes classifier outperforms others classifiers commonly used for pattern recognition.

Table 6: Accuracy by Features.

Features	Precision	Recall	F-Measure
Vertical projection variance (1)	0.43	0.52	0.45
Connected component width, height, aspect ratio, area and density (10)	0.81	0.81	0.81
Separator length between two successive connected components (2)	0.57	0.56	0.56
Connected component profiles analysis (1)	0.61	0.50	0.46
Loop ratio (2)	0.53	0.53	0.52
Pixels distribution (1)	0.32	0.42	0.36
Baseline profile features (8)	0.58	0.84	0.85
Run_length histogram (15)	0.90	0.89	0.89
Crossing count histogram (10)	0.73	0.73	0.72
Word physical sizes (5)	0.58	0.56	0.57
Overlapping areas (1)	0.25	0.29	0.23
Central moments (2)	0.52	0.52	0.51
Hu Moments (2)	0.46	0.48	0.43
Upper_lower profile (3)	0.59	0.59	0.58
Bi-level Co-occurrence (4)	0.75	0.74	0.74
Gabor filters (8)	0.66	0.66	0.66
Steerable pyramid transform (48)	0.89	0.89	0.89

By combining these features with those proposed features in this work, correctly classified instances are 1696 using AODEsr. Only 24 are incorrectly classified. So with these 126 features, words can be identified, in a reliable way, with a correct identification rate of almost 98.60%. The time taken to build model is 0.31 seconds. Table 7 displays the obtained results with different feature selection methods using AODEsr classifier.

It also indicates the time taken to build model. As shown in Table 7, when applying the Genetic algorithm as feature selection method, the selected features are reduced from 126 to 58 features (See Table 8), the correctly identification rate is the highest and the consuming time is among the lowest. As results on Table 9 show, the average accuracy is the same, about 98.72% for handwritten and machine printed words either in Arabic or Latin scripts. Notice that, with the selected features, printed Arabic words can be identified, in a reliable way, with a correct identification rate of 100%. When observing the confusion matrix (See Table 10), we note that it is about confusion cases between handwritten Arabic and Latin scripts. Most of them mainly come from their cursive nature. Confusion, between printed and handwritten Latin script arise because of the writing styles of many writers who do not use ligatures between the letters. We also compared the performance of five typical classifiers: Bayes (AODEsr), *k*-Nearest Neighbor (*k*-NN), Decision Tree (J48), Multilayer Perceptron (MLP) and Support Vector Machine (SVM) (See Table 11). As mentioned before, AODEsr is a probabilistic classification learning technique which was developed to address the attribute-independence problem of the popular Bayes classifier [21]. The *k*-NN is the extension of the Nearest Neighbor classifier which was first introduced by [22]: An unknown word is classified by assigning it the label most frequently represented among

Table 7: Testing with different feature selection methods

Feature selection algorithms	**Selected features**	**Identification Rate** (%)	**Time (s)**
Genetic algorithm	58	98.72	0.03
BestFirst	36	98.43	0.02
PCA & Ranker	48	95.69	0.02

Table 8: Features set after selection.

Feature Set	**Selected Features**
Vertical projection variance	(1/1)
Connected component width, height, aspect ratio, area and density	(2/10)
Separator length between two successive connected components	(2/2)
Connected component profiles analysis	(1/1)
Loop ratio	(1/2)
Pixels distribution	(1/1)
Baseline profile features	(5/8)
Run_length histogram	(9/15)
Crossing count histogram	(2/10)
Hu Moments	(2/2)
Central Moments	(1/2)
Upper_lower profile	(2/3)
Bi-level Co-occurrence	(2/4)
Overlapping areas	(1/1)
Physical sizes	(2/5)
Steerable pyramid transform	(19/48)
Gabor filters	(2/8)
Proposed features	(3/3)

Table 9: Detailed Accuracy by Class.

	PA	**HA**	**PL**	**HL**	**Average**
Precision	0.982	0.991	0.998	0.979	0.987
Recall	1	0.977	0.995	0.977	0.987
F-Measure	0.991	0.984	0.997	0.978	0.987

Table 10: Confusion Matrix.

	PA	HA	PL	HL
PA	430	0	0	0
HA	3	420	0	7
PL	0	0	428	2
HL	5	4	1	420

the *k* nearest word samples. A decision is made by examining the labels of the *k* nearest neighbors and taking a vote. As defined in [23], a decision tree is a predictive machine-learning model that decides the word class of a new word based on various attribute (features) values of the available data. The internal nodes of a decision tree denote the different attributes. The branches between the nodes tell us the possible values that these attributes can have in the observed word samples, while the terminal nodes tell us the corresponding word class. MLP is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate outputs. A MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called back propagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable [25]. Notice that MLPs were a popular machine learning solution, finding applications in diverse fields of pattern recognition, but have faced strong competition from the much simpler SVM which is a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on [26]. As it can be seen, AODEsr provides good results in comparison to the others classifiers.

Table 11: Accuracy by Classifier

Classifier	F-Measure	Time(s)
AODEsr	98.72%	0.03
J48	85.98%	0.14
***k*-NN**	97.5%	0.01
MLP	97.20%	34.34
SVM	97.03%	0.63

In Fig. 24, we display the Receiver Operating Characteristics (ROC) curve to compare the three first classifiers and to highlight what is the classifier that has the best discriminative power. This will be the classifier that has the highest ROC curve widening. Here, it corresponds to AODEsr considering HL class.

5.1 Comparing with Existing System

Using the selected features, the AODEsr classifier captures significant amount of the differences between machine-printed and handwritten Arabic and Latin words providing a good solution for this task. To compare it with a system proposed in [12] which deals with the same problem, we used a common database (our database composed of 1720 word samples). Recall that [12] proposed an identification system, at word level, based on steerable pyramid transform and using *k*-NN as classifier. Notice that we tried to reproduce their system considering the same conditions as stated in their paper [12]. We used steerable-pyramid feature [21] with 4 orientation sub-bands, at 2 scales. Feature vector dimension is 48 to represent 8 sub-bands. The feature vector was constructed using the computed mean, the standard deviation, the Kurtosis, the magnitude of

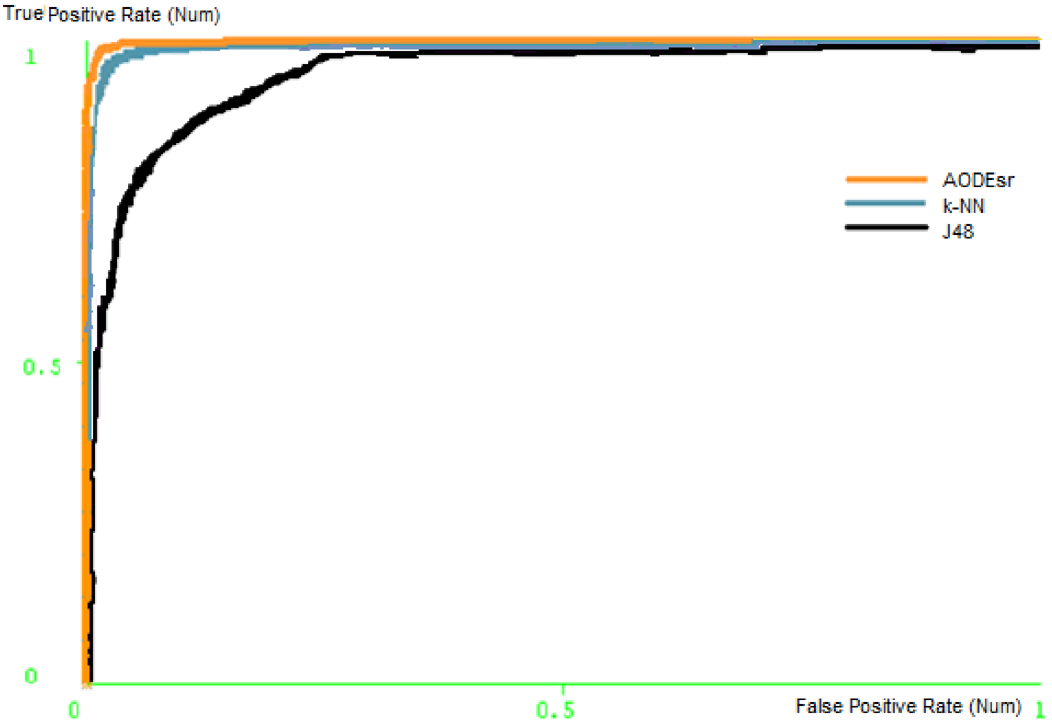


Figure 24: ROC graph for classifier comparison.

the transformed word image and the energy, the homogeneity and the correlation calculated from gray level co-occurrence matrix applied to the same transformed word image. Table 12 summarizes the obtained results. Notice that when using steerable pyramid transform [21] (as feature proposed in [12]) and testing it on a larger database (1720 words instead of 800), the F-Measure is reduced from 97.5% to 93.08%. But the use of AODEsr instead of *k*-NN as classifier has further reduced the rate to 91.86%. In sum, the use of a set of 58 selected features with Bayes classifier achieves an identification rate of 98.72% which is slightly better than 93.08% (the identification rate obtained using features from the steerable pyramid transform with *k*-NN as classifier and tested on the same database). In our view, the obtained results show that giving slightly higher weight to the structural information can produce better results.

Table 12: Comparing with existing system.

System	Feature set	Classifier	Data-Base	F-Measure
[12] tested on its own database	48	*k*-NN	800	97.5%
[21] tested on our database	48	*k*-NN	1720	93.08%
[21] tested on our database with AODEsr	48	AODEsr	1720	91.86%
Our system	58	AODEsr	1720	98.72%

6 Conclusion and perspectives

This paper introduces, classifies, and surveys script and nature identification at word level. Experiments were carried on Arabic and Latin handwritten and machine-printed words. Note that this work is not intended to review methods, but rather surveys results, providing the reader a structure for assessment. In fact, we tried to propose and select features that maximize the distinction between handwritten and machine-printed Arabic and

Latin at word level. A set of 58 simple and different features has been retained after selection. The performances of five classifiers are compared. Handwritten and machine-printed words, with various font styles and sizes, written in Arabic and Latin, have been used for testing the proposed features and classifiers and the results show the identification process is robust and reliable at the word level. Notice that these features may be generalized to include all other Romance and Anglo Saxon languages instead of only English or French and other languages that use Arabic scripts such as Persian and Urdu. In the future, we plan to explore further features and test with broader word databases.

References

[1] A. Kacem, A. Saïdani and A. Belaïd , ”A System for an automatic reading of student information sheets”, *Proc. of ICDAR*, 1265-1269, (2011).

[2] A. Saïdani, A. Kacem and A. Belaïd, ”Identification of machine-printed and handwritten words in Arabic and Latin scripts”, *Proc. of ICDAR*, (2013)

[3] S. Kanoun, I. Moalla, A. Ennaji and A. M. Alimi , ”Script Identification for Arabic and Latin Printed and handwritten Documents”, *Proc. of DAS*, 159-165, (2000).

[4] S. Mozaffari and P. Bahar, ”Farsi/Arabic handwritten from machine-printed words discrimination”, *Proc. of ICFHR*, 694-699, (2012).

[5] L. Faria da Silva, A. Conici and A. Sanchez , ”Automatic discrimination between printed and handwritten text in documents”, *Proc. of SIBGRAPI, XXII Brazilian Symposium*, 261-267, (2009).

[6] L. Zhou , Y. Lu and C. Tan , ”Bangla/English Script Identification based on Analysis of Connected Components Profiles”, *Proc. of DAS*, 243-254,(2006).

[7] M. Benjelil, S. Kanoun, A. M. Alimi and R. Mullot , ”Three decision levels strateg for Arabic and Latin texts differentiation in printed and handwritten natures”, *Proc. of ICDAR*, 1103-1107, (2007).

[8] E. Kavallieratou and S. Stamatatos , ”Discrimination of machine-printed from handwritten Text using Simple Structural Characteristics”, *Proc. of ICPR*, 437-440, (2004).

[9] Y. Zheng , C. Liu and X. Ding , ”Single character type identification”, *Proc. of DRR*, 49-56,(2002).

[10] A. M. Elgammal and M. A. Ismail, ”Techniques for Language Identification for Hybrid Arabic-English Document Images”, *Proc. of DRR*, 1100-1104, (2001).

[11] K. Baâti , Kanoun and M. Benjlaiel , ”Diffrenciation d’écriture Arabe et Latine de natures Imprimée et Manuscrite par approche globale”, *Proc. of CIFED*, 313-324, (2010).

[12] M. Benjelil , R. Mullot and M. A. Alimi, ”Language and script identification based on Steerable Pyramid Features”, *Proc. of ICFHR*, 18-20 September, Bary-Italy, 712-717, (2012).

[13] S. Haboubi , S. S. Maddouri and H. Amiri , ”Discrimination between Arabic and Latin from bilingual documents”, *Proc. of CCCA*, (2011).

[14] G. G. Rajput and H. B. Anita , ”Handwritten Script Identification from a Bi-Script Document at Line Level using Gabor Filters”, *Proc. of SCAKD*, 94-101, (2011).

[15] Y. Zheng , H. Li and D. Doermann , ”Machine-printed text and handwritten identification in noisy documents images”, *IEEE PAMI*, 26(23):337–353, (2004).

[16] H. Ma and D. Doermann , ”Word level script identification for scanned document images”, *Proc. of DRR(SPIE)*, 124-135, (2004).

[17] R. O. Duda , P. E. Hart and D. G. Stork, ”Pattern classification”, *A Wiley-Interscience publication*,(2001).

[18] B. W. Cavnar and J. M. Trenkle, ”N-gram based text categorization”, *Proc. of SDAIR*, 161-175, (1994).

[19] D. E. Golberg , ”Genetic algorithm in search, optimization and machine learning”, *Addison-wesley Longman Publishing*, (1989).

[20] http://wiki.pentaho.com/display/DATAMINING/BestFirst

[21] D. Nguyen, ”Simplified Steerable pyramid”, available at: www.mathworks.com/matlabcentral/fileexchange/36488-simplified-steerable-pyramid, (2012).

[22] T. M. Cover and P. E. Hart, ”Nearest Neighbor Pattern Classification”, *IEEE Trans. Information Theory*, 13(1), 21-27, (1967).

[23] ”Classification methods”, available at: http://www.d.umn.edu/ padhy005/Chapter5.html.

[24] S. Haboubi, S. Snoussi Maddouri and N. Ellouze, ”Différenciation de documents textes Arabe et Latin par filtre de Gabor”, *Proc. of TAIMA*, (2007).

[25] R. Collobert and S. Bengio, ”Links between Perceptrons, MLPs and SVMs”, *Proc. of Int’l Conf. on Machine Learning*, (2004).

[26] C. Cortes and V. Vapnik, ”Support-vector networks”. *Machine Learning*, 20(3), 273-297, (1995).

[27] K. Zagoris, I. Pratikakis, A. Antonacopoulos, B. Gatos and N. Papamarkos, ”Distinction between handwritten and machine-printed text based on the bag of visual words model”, *Pattern Recognition*, 47(3), 1051-1062, (2014).

[28] U. Pal and B. B. Chaudhuri, ”Machine-Printed And Hand-Written Text Lines Indentification”, *Pattern Recognition Letters*, 22(3-4), pp. 431-441, (2001).

[29] X. Peng, S. Setlur, V. Govindaraju and R. Sitaram, ”Handwritten text separation from annotated machine printed documents using Markov Random Fields”, *International Journal on Document Analysis and Recognition (IJDAR)*, 16(1), pp 1-16, (2013).