Arabic/Latin and Machine-printed/Handwritten Word Discrimination using HOG-based Shape Descriptor

Asma Saïdani*,Afef Kacem Echi* and Abdel Belaïd+

* *University of Tunis, ENSIT-LaTICE, 5 Avenue Taha Husseïn, BP 56 Babmnara 1008 Tunis, Tunisia*

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

Abstract

In this paper, we present an approach for Arabic and Latin script and its type identification based on Histogram of Oriented Gradients (HOG) descriptors. HOGs are first applied at word level based on writing orientation analysis. Then, they are extended to word image partitions to capture fine and discriminative details. Pyramid HOG are also used to study their effects on different observation levels of the image. Finally, co-occurrence matrices of HOG are performed to consider spatial information between pairs of pixels which is not taken into account in basic HOG. A genetic algorithm is applied to select the potential informative features combinations which maximizes the classification accuracy. The output is a relatively short descriptor that provides an effective input to a Bayes-based classifier. Experimental results on a set of words, extracted from standard databases, show that our identification system is robust and provides good word script and type identification: 99.07% of words are correctly classified.

Key Words: Script and type identification, Histogram of oriented Gradients, Arabic and Latin separation

1 Introduction

Nowadays, we see more and more in the transferring documents circuits, documents with mixture of scripts, as well as type of mixture: printed and manuscript. This is due to some annotations or signatures quickly affixed by the writers to certify their documents or simply because of the bilingualism of certain companies or administrations. For document analysis and recognition systems, this requires to add specific pre-processing to separate these different types of scripts to orient them to dedicated processing modules. As a contribution to this momentum concerning the separation of Latin/Arabic scripts, we propose to investigate the suitability of Histogram of Oriented Gradient (HOG) and two of these extensions : Pyramid HOG and Co-occurrence HOG. Being based on shape descriptors, they have interesting properties for script characterization.

The literature abounds with identification scripts techniques using different features at block, text-line or word levels. But most of these features are not effective to capture differences between handwritten Arabic and

Latin scripts especially due to their cursive nature. In this paper, we try to exploit the writing direction sensitive as a discriminative aspect for these script separation.

For this, we rely on the observation that we made in the examination of the morphology of these two scripts: Latin and Arabic. Arabic, as written from right to left, letters in words (especially of handwritten type or italic machine-printed) are generally tilted to the left, following the writing direction (see Fig. 1(a)). In contrast, Latin script, as written from left to right, letters (especially of handwritten type or in italic machine-printed) tend to be inclined to the right (see Fig. 1(b)). Thus, Arabic letter strokes are generally diagonally down whereas those written in Latin 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. 1(c)). In contrast, machine-printed Latin words are composed by successive letters without any ligature between them (see Fig. 1(d)). Thus, horizontal strokes would be more frequent in Arabic words than in Latin 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 type identification.

(a) Handwritten Arabic   (b) handwritten Latin

(c) Machine-Printed Arabic   (d) Machine-Printed Latin

Figure 1: Arabic and Latin handwritten and machine-printed words.

The paper is organized as follows. We firstly give, in section 2, a brief description of Arabic and Latin script properties. We then discuss, in section 3, some related works. In section 4, we precisely define basic HOG descriptors. Afterwards, we propose in section 5 many improvements of the HOG for the specific task of machine-printed/handwritten and Arabic/Latin word discrimination and compare them experimentally in section 6. We finally state, in section 7, the conclusions and give perspectives.

2   Arabic/Latin Differences

Arabic and Latin have a completely different typography regarding their respective use of the Cartesian space. Whereas the Latin script divides the used space in roughly three different areas, the shapes of the Arabic script in its simplest form can be grouped in five areas on the vertical axis. This results in more differentiation of shapes and accordingly to a more varied and moving pattern in text. Further the wide range of ascending and descending elements is accompanied by relatively small shapes and counters along the joining line of the letters. These elements constitute inherent characteristics of the Arabic script (see Fig. 1(a) and (c)).

Arabic is also different from Latin with respect to a number of aspects. It is right to left for Arabic and from left to right for Latin. 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 (see Fig. 1(b) and (d)). 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. 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. Letters of a PAW could be horizontally or vertically tied (in some fonts, two, three or even four letters can be tied).

The same Arabic word does not have a fixed length since different elongations numbers could appear between letters. The foregoing features relate to the Arabic writing whether printed or handwritten.

3   Literature Review

There is an abundant literature on script identification. We mention in the following some relevant papers related to this aspect for Arabic/Latin languages at block, text-line, word and connected component levels (see Table 1).

Most of used features are statistical such as connected component width, height, aspect ratio, area and density and separator length between two successive connected components [13], connected component profiles analysis [10], loop aspect ratio [1], pixel distribution and upper lower profile by [3], baseline profile features, word physical sizes, overlapping areas, crossing count histogram [2], bi-level co-occurrence [9], run-length histogram [4], Gabor filters [6] and [38], steerable pyramids transform [7] or structural (ascenders, descenders, loops, diacritic signs, etc.).

The vectors associated with these features are often used as input for word classifiers such as K-NN or SVM. In this section, we only focus on works based on features sensitive to the direction. We then explore the use of HOG-based shape descriptors for Arabic/Latin and handwritten/machine-printed at word level.

Table 1: Identification systems summarization

| Reference	Script	Type (number)	Level	Identification rate
[3]	L	P/H (50D)	T	98.2%
[12]	L	P/H (1539D)	T	100%
[2]	F	P/H (32006W)	W	97.1%
[10]	G/L	H (1200B)	C	95%
[1]	A/L	P/H (400B)	B	88.5%
[38]	A/L	P/H (400B)	B	95%
[6]	A/L	P/H (800D)	B	84.75%
[11]	A/L	P/H (600D)	D	82%(A), 92%(L)
[5]	A/L	P (1976T, 8320W)	T/W	99.7%(T), 96.8%(W)
[7]	A/L	P/H (800W)	W	97.5%
[8]	A/L	P (4229W)	W	94.32%
[23]	A/L	P/H (1720W)	W	98.72%
[15]	A/L	H (1068W)	W	100%(L), 98.88%(A)
[28]	A/L	P/H (20000W)	W	99.10%
[30]	A/L	P/H (2000W)	W	98.02%

A:Arabic, B:Block, C:Connected component, D:Document, F:Farsi_Arabic, G:Bangla, H:Handwritten, L:Latin, P:Printed, T:text-line, W:Word

Zheng et al. [4] proposed features from Run-length histogram for machine-printed/handwritten Chinese character classification. Black pixel run-lengths are extracted in three directions: horizontal, vertical and diagonal. Three histograms of run-lengths in these directions are then performed. To get scale-invariant features, the histograms are normalized. Then, each histogram is divided into five bins with equal width, and five rectangular-shaped weight windows are used. Thus, five features are extracted in each direction.

Table 2 presents the black vertical run-length of some words. We noted that run length values are high for handwritten words. We think that these features can be used to underline the difference between the stroke length of machine-printed and handwritten words.

To discriminate handwritten from machine-printed words in Farsi Arabic documents at word level, authors in [2] extracted three different sets. The first set includes structural features related to the physical sizes of the blocks (density, width, height, aspect ratio and area), features based on connected components inside the blocks (mean and variance of connected component width, height, aspect ratio and area) and the overlapping area, normalized by the total area of the block. The second set includes crossing count histogram used to measure stroke complexity. Features of the third set are extracted from baseline profiles. Recall that crossing

Table 2: Black vertical run length histogram.

Machine-printed Arabic	Handwritten Arabic
ميزانية	متكافئ
Machine-printed Latin	**Handwritten Latin**
algorithme	simples
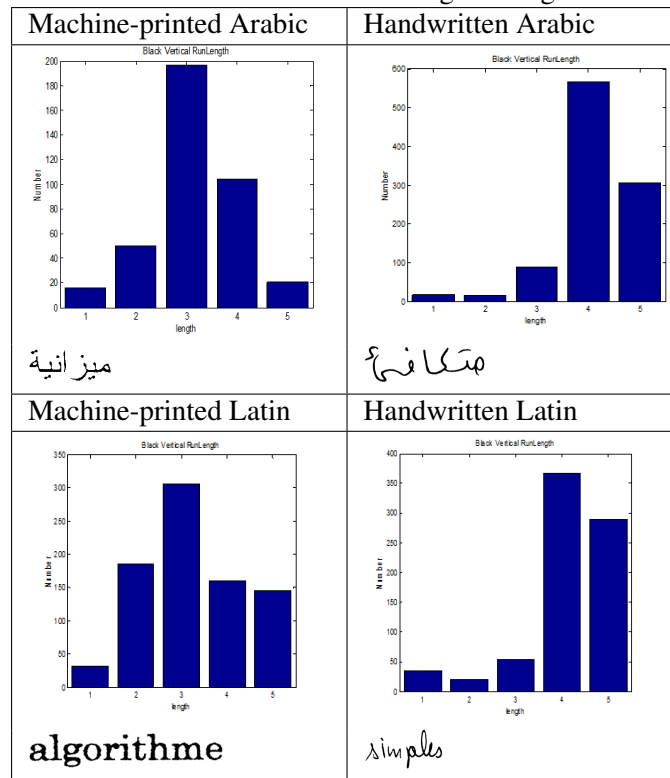
count is the number of transitions from 0 to 1 along a hypothetical horizontal or vertical line over the word image. For each horizontal and vertical scan line, the crossing count is calculated. Two histograms have been got for the horizontal and vertical crossing counts respectively. To have the final features from the histograms, the same technique, used to extract the run-length features, is exploited. We think that these features are useful to measure stroke complexity and can be discriminative for script identification.

Authors in [9] used bi-level co-occurrence technique for machine-printed and handwritten text identification in nosy documents images. A co-occurrence count is the number of times a given pair of pixels appears at a fixed distance and orientation. In binary image, the eventual co-occurrence pairs are white-white, black-white, white-black and black-black. As the most of the information in the image is represented by the black-black pairs, authors have only considered them to extract related features. Horizontal H, vertical V, major diagonal MD and minor diagonal mD orientations and 2 pixels distance d level for the classification are used (see Table 3).

Table 3: Bi-level co-occurrences

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

Authors in [7] proposed a script identification system, at word level, based on steerable pyramid transform. A 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. Notice that steerable pyramids have demonstrated discrimination properties for texture characterization, and unlike other image decomposition methods, the feature coefficients are less modified under the presence of image rotations or even scales. For script identification, features are extracted from pyramid sub-bands and served to classify the scripts. Authors used steerable-pyramid feature with 4 sub-band orientations, at 2 scales (see Fig. 2). Feature vector dimension is 48 to represent 8 sub-bands. The feature vector was constructed by computing the mean, the standard deviation and the Kurtosis of the magnitude of the transformed word image. The energy, the homogeneity and the correlation are calculated from gray level co-occurrence matrix applied to the same transformed word image.

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

Gabor filters have been found to be particularly appropriate for texture representation and discrimination. Authors in [13] applied Gabor filters and 16 channels of features are extracted to identify the script (English or Chinese) of machine-printed words in scanned document images.

To differentiate between Arabic and Latin scripts in printed or handwritten documents, Haboubi et al. [38] used Gabor filters to extract features in the form of a vector of dimension 32. Authors in [6] used a global approach and tested Gabor filters, gray-level co-occurrence matrices and wavelets separately. The energy of the output image after application of a bench of Gabor filters on an original document is always regarded as a characteristic being able to identify the script even in the presence of small base of documents. The gray-level co-occurrence matrices define the relations between the different gray levels of an image. The highest rate of correct identification is obtained with gray-level co-occurrence matrices. In previous work [23], we tested all of the above script identification techniques to discriminate between Arabic/Latin and machine-printed/handwritten word. We present the obtained results for each technique on a database of 1720 words. A comparative study was performed to compare between the techniques and an attempted combination of features has been proposed to improve results.

we cite now the most recent work that addresses the same issue, which are three works: [15], [30] and [28].

In [15] present a performance comparison of curvelet, dual-tree complex wavelet and discrete wavelet transform in handwritten words classification (Arabic and Latin). A database of 1086 handwritten Latin (HL) and Arabic (HA) word images are used, 543 for training and 543 for testing. This database is constructed from two publics database : IFN/ENIT-database [25] for Arabic handwritten words and IAM-database [26] for Latin handwritten words. The feature vector was built using the energy and entropy calculated at different decomposition sub-bands. Support vector machines (SVM) and K-NN are used for classification. The experiments showed that the best identification results are reached using curvelet transform.

In our former work [30], we used pyramid histogram of oriented gradients (PHOG) to discriminate Arabic/Latin and handwritten/printed words. PHOG features, counts occurrences of gradient orientation in localized portion of an image. It has been showed as an effective tool for supplying spatial distribution of pixels. The feature vector of an image is composed of 129 values after selection with genetic algorithm. Experiments have been conducted using a database of 2000 words. An average identification rate of 98.3% was achieved

using Bayes based classifier.

Different features are considered by [28] for the classification of a given word image into one of four classes: Arabic-Handwritten, Arabic-Printed, French-Handwritten and French-Printed. The word image is normalized to a size of 30 pixels height. Then it is transformed into a sequence of feature vectors computed from a fixed-length analysis window sliding from right to left over the word image. Each window is represented with a 35 features whose distribution is captured by Gaussian Mixture Models (GMMs). These features include affine moment invariants, the number and the XY position of the top and the bottom extrema, cumulated horizontal projection values at 10 equal parts divided by the height of the image, the maximal amplitude obtained from the difference between the top and the bottom profiles at particular locations divided by the height of the image. Notice that GMMs are well-known multipurpose and powerful modeling tools able to model thinly the distributions of features that are quite wide-scoped and largely multi-dimensional. Experiments used three public databases of Arabic and French: AHTID/MW [27], the APTI database [32] (composed of low-resolution (70 dpi) and synthetically generated images) and RIMES [21]. These experiments showed a remarkable performance.

4   Fundamentals of gradient-based image descriptor

In this section we provide a description of the HOG, PHOG and Co-HOG descriptors.

- **HOG:** The use of HOG as texture decriptor was introduced by Dalal and Triggs[19] for human recognition. HOG descriptor is also used in some recent text recognizers [29] and for recognition of handwritten words by [34].

  The main idea of HOG is that the shape of objects in an image can be characterized by the distribution of local intensity gradients representing the dominant pixel directions. A HOG descriptor is a histogram which counts the gradient orientation at pixels in a given image. We briefly explain the essence of HOG calculation process in Fig. 3. In order to extract HOG from an image, firstly gradient orientations at every pixel are calculated (see Fig. 3(a)). Secondly, a histogram of each orientation in a small rectangular region is calculated (see Fig. 3(b)). Finally, A HOG feature vector is created by concatenating the histogram of all small regions (see Fig. 3(c)). The result of the HOG algorithm is a discrete amount of features which describe the input image. The number of features depends on the number of cells and orientation. In next section, we explore several ways to enhance HOG at minimal performance cost.

- **PHOG:** Pyramid HOG takes the spatial property of the local shape into account while representing an image by HOG. The spatial information is represented by tiling the image into regions at multiple resolutions based on spatial pyramid matching (see Fig. 11). Each image is divided into sequence of increasingly finer spatial grids by repeatedly doubling the number of divisions in each axis direction. The number of points in each grid cell is then recorded. The number of points in a cell at one level is simply the sum over those contained in the four cells it is divided into at the next level thus forming a pyramid representation as explained by [17]. PHOG achieved good performance in many studies by: for the retrieval of historic Chinese architectures image [33], to classify images by the object categories they contain [17] and for smile recognition [22].

- **Co-occurrence HOG:** the Co-occurrence HOG is an extension of HOG that considers relation between pairs of pixels, proposed by [37] for human detection. Co-occurrence HOG based features have been applied for face recognition [35], for scene text recognition [31] and for character recognition in natural scenes [16]. In fact, this feature captures spatial information by counting the frequency of co-occurrences of oriented gradients between pairs of pixels. Thus relative locations are stored. The relative locations are reflected by the offset between two pixels as shown in Fig. 4(a). The offset $(\triangle_x, \triangle_y)$ specifies the distance between the pixel of interest and its neighbor. The yellow pixel in the center is the pixel under

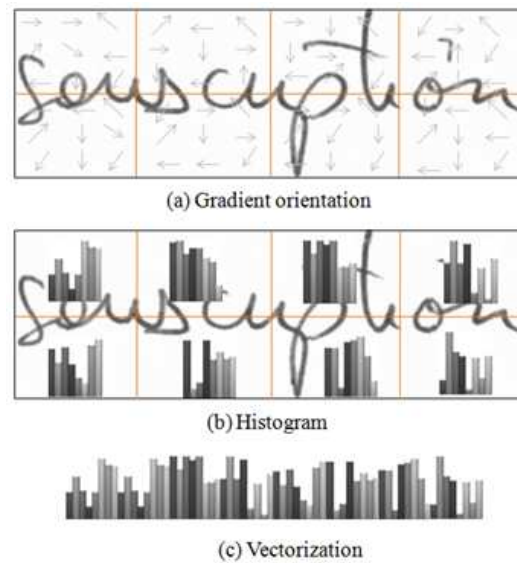(a) Gradient orientation

(b) Histogram

(c) Vectorization

Figure 3: Overview of HOG calculation.

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. 4(b). Therefore, HOG is just a special case of Co-HOG when the offset is set to (0; 0), that is only the pixel under study is counted. The frequency of the co-occurrences of oriented gradients is captured at each offset via a co-occurrence matrix as shown in Fig. 4(b). Compared with HOG, Co-HOG captures more local spatial information but at the same time keeps the advantage of HOG, i.e., the robustness to illumination variation and invariance to local geometric transformation.

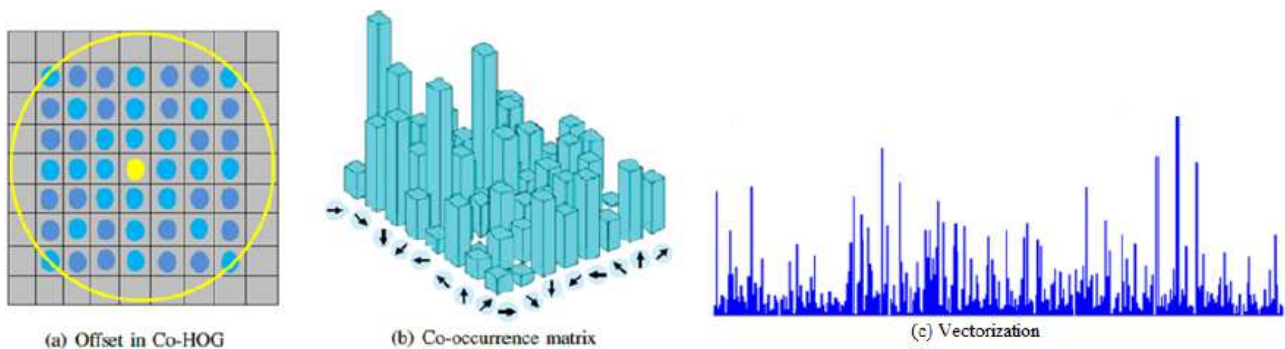(a) Offset in Co-HOG   (b) Co-occurrence matrix   (c) Vectorization

Figure 4: Illustration of Co-HOG features extraction: (a) offset in Co-HOG, (b) Co-occurrence of a word image at a given offset, (c) Vectorization of co-occurrence matrix [36].

In next section, we precisely define the application of HOG, PHOG and Coo-HOG descriptors to word images with prior improvements.

5   Features Extraction

Features extraction is the most important step of any pattern recognition system which aims to find an appropriate representation of the pattern under study. In our case, gradient features were applied to the machine-printed/handwritten and Arabic/Latin word discrimination. To take full advantage of the gradient features, we

propose to apply the HOG descriptor using two different techniques : Pyramid HOG and Co-occurrence HOG. The compute of gradient orientations contains some steps which are common in the two aforementioned techniques.

- **Step 1:** The horizontal and vertical gradients of the image is computed by applying the 1D centered point discrete derivative masks with the following filter kernels:

$$D_x = \left(\begin{array}{ccc} -1 & 0 & 1 \end{array} \right) \ and \ D_y = \left(\begin{array}{c} 1 \\ 0 \\ -1 \end{array} \right) \tag{1}$$

So, given an image I (see Fig. 5(a)), we obtain the x and y derivatives using a convolution operation: $R_x = I * D_x$ and $R_y = I * D_y$ as shown in Fig. 5(b) and Fig. 5(c).

(a) Initial image

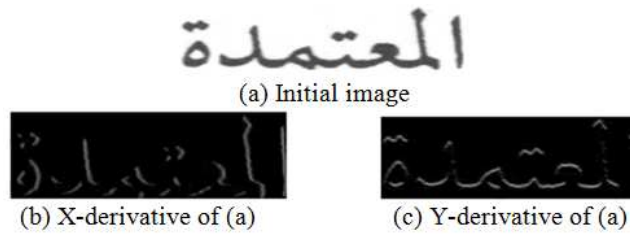(b) X-derivative of (a)   (c) Y-derivative of (a)

Figure 5: Example of Gradient Computation.

- **Step 2:** The magnitude and orientation of the gradient are computed using Eq. 2 and Eq. 3.

$$R_g = \sqrt{R_x{}^2 + R_y{}^2} \tag{2}$$

$$R_o = \arctan \frac{R_y}{R_x} \tag{3}$$

Where: R_g and R_o denote the magnitude and orientation respectively of an image.

- **Step 3:** This step in calculation involves creating the cell histogram. Each pixel within the cell casts a weighted vote for an orientation-based histogram channel based on the values found in the gradient computation. The histogram channels are evenly spread over $0°$ to $180°$ or $0°$ to $360°$ depending on whether the gradient is "unsigned" or "signed" (from $0°$ to $360°$ in our case). The best value of the number of bins of the histogram is 8. This will be confirmed in our experiments by testing with different values of bin number. The histogram of orientation is calculated at each bin as follows:

$$H_i = \sum_{x,y \in I, R_o(x,y) \in bin_i} R_g(x,y), i = 1, .., n \tag{4}$$

In experiments, the interval $[0, 2\pi]$ is divided into $n = 8$ bins. Each bin covers an orientation range of $\frac{\pi}{4}$.

- **Step 4:** The histogram of oriented gradients is normalized by L2-Norm represented as follows: $v \to \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}}$

5.1 Muti-cell HOGs

We first propose to use HOG with a specific cell layout which turns out to be a division of word image into horizontal and vertical strips with and without overlapping. In this sense, we show that the flexibility of the grid improves the results of the rigid grid that the HOG uses. Notice that images of words have distinct HOGs

in different parts (top, middle and bottom) especially when the words have ascending and descending letters (see Fig. 6). We can see that when we divide a word image into three equal horizontal strips, the bottom parts of the Arabic words contain a larger proportion of horizontal strokes which refer to elongate descenders. Consequently, the gradients in these parts are mostly vertical as shown in Fig. 6(a) and (c). Also, the top parts of words contain a larger proportion of vertical strokes which correspond to the ascenders. Thus, the gradients in these parts are mostly horizontal (see Fig. 6(a) and (b)). The middle parts of machine-printed Arabic and Latin words are almost the same with a larger proportion of vertical strokes which correspond to central letters, such as Arabic letters: ـس, ـبـ, ـنـ, ـبـ, etc. and Latin letters: n, m, r, etc.), lower parts of ascender letters (such as

parts of Latin letters: l, t, etc.) or upper parts of descending letters, such as part of Arabic letter (ل, etc.). So, the gradients in these parts are almost equally distributed but mostly horizontal (see Fig. 6(a) and (b)). In the middle parts of handwritten Arabic and Latin words and due to their cursive nature, there is a larger proportion of diagonal strokes which are almost uniform over the whole range $[0, \pi]$. This is due to the rounded parts of letters (such as Latin letters: O, D, e, d, a, and Arabic letters: ـهـ, ـة and ة etc. as shown in Fig. 6(c) and (d)).
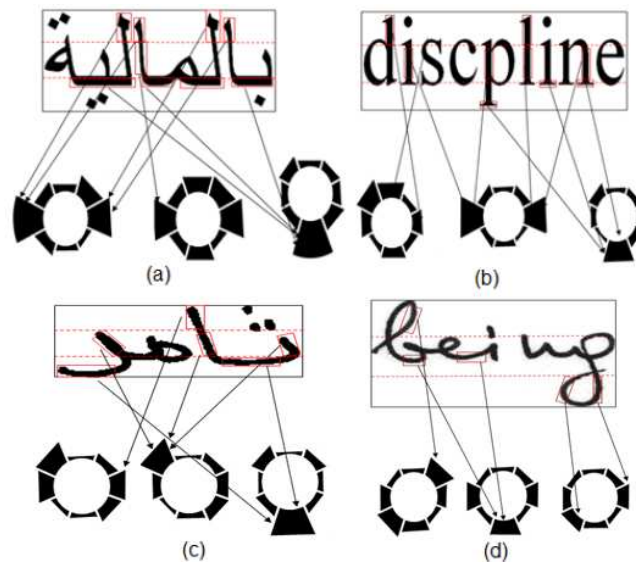

Figure 6: From left to right, the 8_bin HOG descriptors of the top, middle and bottom parts of (a) printed Arabic word, (b) printed Latin word, (c) handwritten Arabic and (d) handwritten Latin word. The arrows indicate the contribution of specific letter strokes to the histogram.

As not all of Arabic and Latin words have ascending and descending letters, we next performed a series of tests to determine the optimum cell arrangement for script and nature identification at word level with HOG algorithm which also allows the cells to be grouped into blocks which may partially overlap.

- **First arrangement (Central band based division):** In this case, we try to take advantage of image content which is writing. We proceed by division according Observing the differences in the orientation distribution in the top, middle and bottom parts of word images, we proceed by dividing words into three bands (upper, central and lower bands, see Fig. 7), using horizontal projection, and computed a HOG for each band separately, then concatenated those HOGs to obtain a composite descriptor.

  Notice also that the central band contains the most amount of information, we propose an other arrangement based in the first one by segmentation of the middle part into five equal cells (see Fig. 8). So, we extracted seven HOGs from the word image(5 HOGs from the central band plus 2 HOGs from the upper and lower bands).

- **Second arrangement (Cells arrangements without overlapping):** We also tried to vary the number of

Figure 7: Central band based division.

Figure 8: Dividing word central band into 5 cells.

cells in the horizontal (n_x) and vertical axis (n_y)with the HOG algorithm, as a function of the total cell count $n_x * n_y$. The possible arrangements with six cells are displayed in Fig. 9.
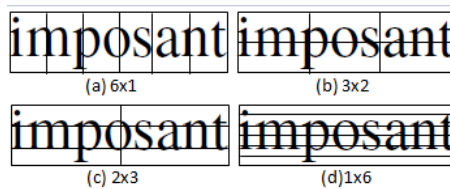
(a) 6x1 (b) 3x2

(c) 2x3 (d)1x6

Figure 9: Cell arrangements without overlapping: (a) 6 columns*1 row, (b) 3 columns*3 rows, (c) 2 columns*3 rows and (d) 1 column*6 rows.

- **Third arrangement (Cell arrangements with overlapping):** As mentioned before, HOG allows the cells to be grouped into blocks which may partially overlap. The possible arrangements with six cells (counting overlaps) are shown in Fig. 10.

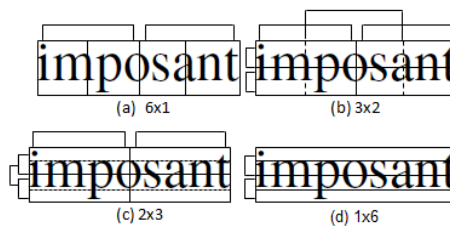(a) 6x1 (b) 3x2

(c) 2x3 (d) 1x6

Figure 10: Cell arrangements with overlapping.

to compute the feature vector from an image using Multi-cell HOGs, a histogram of orientations is computed for each cell independently. the final Multi-cell HOGs descriptor is the concatenation of all vectors of different cells.

5.2   Pyramid HOG

To apply PHOG we start by computing the HOG for the entire image (see Fig. 11(a)). The bin size for HOG is fixed throughout as n bins (here $n = 8$). Each bin in the histogram represents the number of pixels that have

an orientation within a certain angular range. Then, the image is divided into four areas and HOG is computed for each one of them (see Fig. 11(b)). This process is recursively repeated up to a depth L (see Fig. 11(c) and (d)). Consequently, the word image is divided into cells at several pyramid levels. Level 0 is represented by an n-vector corresponding to the n bins of the histogram. Level 1 is represented by a 4 n-vector and the PHOG descriptor of the entire image is a vector with dimensionality equals to:

$$K * \sum_{l=0}^{L} 4^l \tag{5}$$

The histograms of the same level are concatenated into one vector. The final PHOG descriptor for an image is a concatenation of all vectors at each pyramid resolution (i.e. level). In our case, we have set $n = 8$ and $L = 3$. Hence, our PHOG descriptor is of length $8 * (1 + 4 + 16 + 64) = 680$.
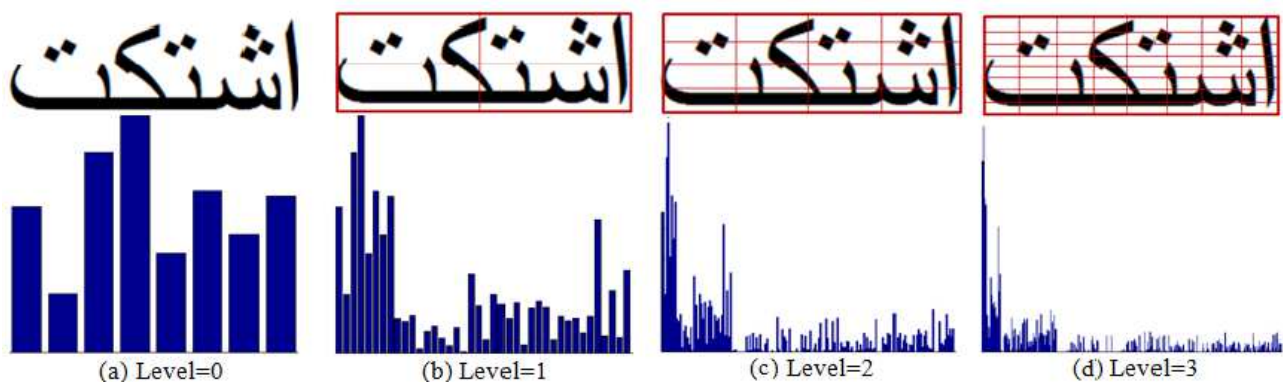

Figure 11: A schematic illustration of PHOG at each resolution level.

Fig. 12 shows the rectangular and circular PHOG descriptors at first level, which corresponds to one cell (basic HOG) of four handwritten Arabic and Latin word images. The displayed HOGs have 8 bins, each bin $\frac{\pi}{4}$ radians wide. As it can be noted, machine-printed Arabic and Latin word images have similar HOG representations (see Fig. 12 (b) and Fig. 12 (d) whereas handwritten Arabic word images have HOG representation far enough from the Latin ones (see Fig. 12 (a) and Fig. 12 (c)). One can see that HOG gives the predominant orientation of the letter strokes. For example, the histogram of Arabic handwritten words has significant spikes over the range $\left[\frac{\pi}{2}, \frac{3\pi}{4}\right]$ which follows the Arabic writing orientation (from right to left) as shown in Fig. 12 (a). On the contrary, the histogram of Latin handwritten words has significant spikes over the range $\left[\pi, \frac{5\pi}{4}\right]$ which also follows the Latin writing orientation (from right to left) as shown in Fig. 12 (c). Note that PHOG, at first level, can be sufficient for handwritten Arabic/Latin discrimination. But it seems to be unable to separate between machine-printed Arabic and Latin words. The recourse to a higher level analysis is required. That is why PHOG is used.

As it will be shown, basic HOG, multi-cell HOGs, with and without overlapping, and PHOG seem to be efficient to separate machine-printed from handwritten Arabic and Latin words. But, magnitude and orientation gradients are computed at only pixel level. That is spatial information (the relationship) between pairs of pixels is not taken into account. We think that if some relationships between pixels are considered, results will be better. This information can be extracted from a co-occurrence HOG.

5.3   Co-occurrence HOG

For a given image I, of size $N * M$ where R_o is its gradient orientation matrix, the Co-HOG matrix P can be defined as follows where the offset $(\triangle_x, \triangle_y)$ specifies the distance between the pixel of interest and its neighbor.
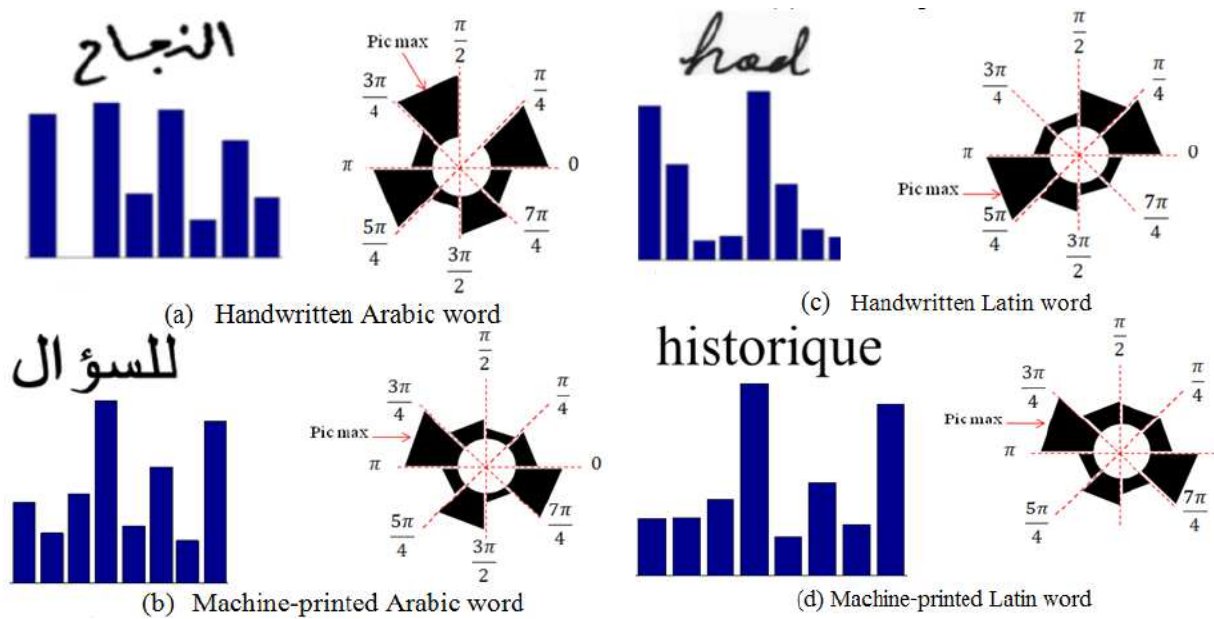
Figure 12: The 8-bin HOG descriptors (at first level) of handwritten/machine-printed and Arabic/Latin word images.

$$P(i,j) = \sum_{x=1}^{N} \sum_{y=1}^{M} \begin{cases} 1, & if\ R_o(x,y) = i\ and\ R_o(x + \triangle_x, y + \triangle_y) = j \\ 0, & otherwise \end{cases} \tag{6}$$

Note that P is a square matrix and its dimension is $n * n$ where n is the number of orientation bins (here $n = 8$ bins). The normalized Co-HOG matrix is denoted by $P_{ij}(d, \theta)$ where d is the displacement vector and θ is the angle. A multiple pair comparisons of tests are performed to see if all distances are individually important or they can be grouped into different categories (such as "near" when distance varies from one to three pixels from the pixel of interest and "far" when distance is more than three pixels). Thus, the numbers of Co-HOG matrices calculations could be significantly reduced. To further reduce the calculations, we only compute Co-HOG on top half of reference pixel, considering just 4 angles (see Fig. 13(a)). So only four orientations are considered in computing co-occurrence HOG matrices : 0°, 45°, 90°and 135°. In Arabic and Latin scripts (especially of machine-printed nature) there are many letters which are simply composed of vertical strokes (e.g. the Arabic letters: letters: l, t, see Fig. 14(a)). In machine-printed Arabic, letters are horizontally ligated and ligatures may be more or less long according to the used font (some fonts allow letter elongations, see Fig. 14(a)). So for these letters and horizontal ligatures, many neighboring pixels have the same directions. In these cases, a "far" distance could be enough to characterize letter strokes without loss of information since there is no morphological variation. As against, in the case of cursive writing (Arabic and handwritten Latin scripts), where letters are not necessarily rectilinear (e.g. 'r', 'e', 'c', 'o', etc. see Fig. 14(b), the use of a "near" distance is obvious to better describe letter shapes (e.g. bends, loops, angles, stems, legs, etc.). One should choose the best distance which considers any morphological change in letters without consuming a lot of time for Co-HOG matrices calculation.

5.4   Co-occurrence and Pyramid of HOG (Co-PHOG)

In this section, based on experimental results we propose a combination between PHOG and Co-HOG in order to benefit from their advantages: 1) the counts occurrences of gradient orientation in localized portion of an image (from PHOG) and 2) the spatial distribution of neighboring orientation pairs (from Co-HOG) instead
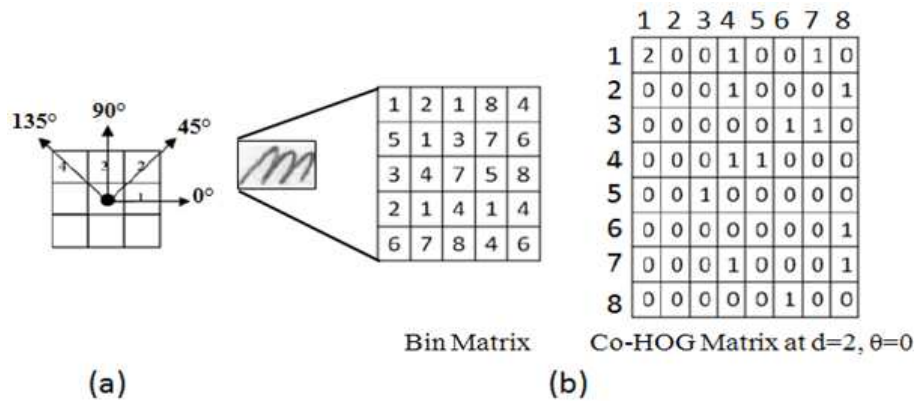
Figure 13: (a) From the centered pixel (o); pixel 1 represents 0 degrees at $d = 1$; pixel 2 represents 45 degrees; pixel 3 represents 90 degrees and pixel 4 represents 135 degrees at $d = 1$; (b) Co-HOG matrix calculated at $d = 2, \theta = 0$ degrees for a 8 by 8 neighborhood around the pixel of interest.

Figure 14: (a) Simple shape letters, (b) Complex shape letters.

of just a single gradient orientation. We call the newly proposed local shape descriptor Co-PHOG. Fig. 15 explains how Co-occurrence matrices could be combined with PHOG.
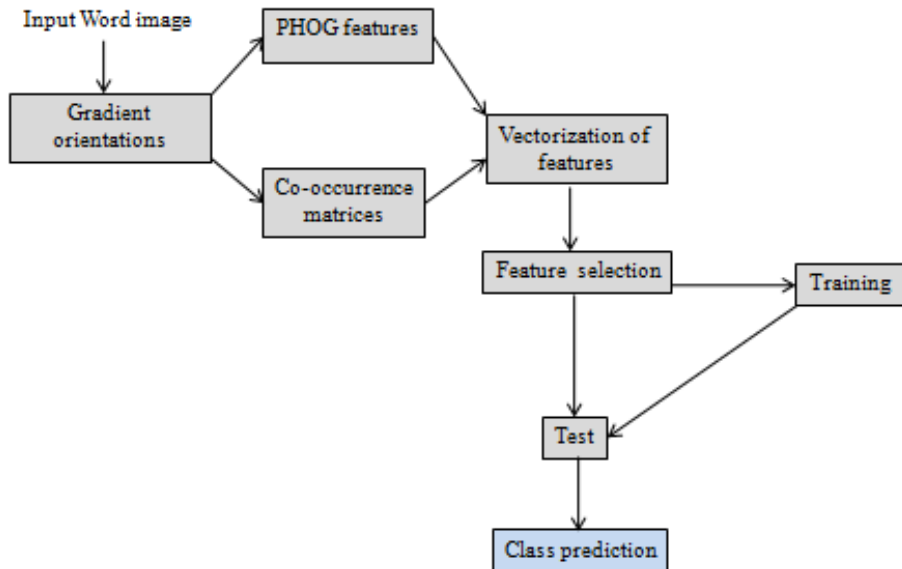

Figure 15: Combining co-occurrence and pyramid of HOG (Co-PHOG)

First, we separately compute and vectorize PHOG and Co-HOG features, as explained above. Then, we concatenate them one after another to form Co-PHOG features vector. As the dimension of the features vector is relatively high (680 from PHOG and 256 from co-occurrence HOG), we tested with different features selection algorithms such as Genetic Algorithm (GA) and Principal Component Analysis (PCA) as dimension reduction methods to alleviate the problem. In fact, smaller descriptors are interesting, even if less accurate, because

they are more computationally efficient and may help us to identify the aspects of the word image that are most relevant for machine_printed/handwritten and Arabic/Latin word discrimination. We finally use a classifier after a training step. We tested with different classifiers as it will be shown later to achieve the better identification rate.

6 Experimental Results Analysis

We made a detailed study of the effects of various implementation choices (mainly about cell arrangements, number of features, number of orientation bins and Co-HOG distances) on script identifier performance, taking machine-printed/handwritten and Arabic/Latin word discrimination as a test case.

6.1 Database

We consider here word images from two public databases: IAM database for Latin handwritten [26] and IFN-ENIT [25] for Arabic handwritten words. For Latin and Arabic machine-printed scripts, we build our own database by extracting words from various magazines and newspapers which contain variable font styles and sizes. A scanning resolution of 300 dpi is employed for digitization of all the words. The training and test words have 4000 samples, 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.

6.2 Experiments

For simplicity and speed, we use a Bayes-based classifier throughout the study. To test the performance of a classifier, we use K-fold cross-validation and leave one out methods. Cross-validation is then repeated K times with all of the K sub-samples used exactly once as the validation data. The cross-validation estimation of accuracy is the overall number of correct classification divided by the number of instances in the data set. In our case, K is set to 10.

As the dimension of the HOG-based shape descriptors is relatively high, we use a genetic algorithm as a dimension reduction method to alleviate the problem. In fact, smaller descriptors are interesting, even if less accurate, because they are more computationally efficient and may help us identify the aspects of the word image that are most relevant for machine-printed/handwritten and Arabic/Latin word discrimination.

6.3 Effect of cell arrangements

We tested many possible cells arrangements with and without overlapping blocks as shown in Tables 4, 5 and 6. Time sums times needed for features selection and word classification (for all 4000 words).

Table 4: Identification rate versus cells arrangements (Central band division)

Cells number	3	7
Features number	24	56
Selected features	14	31
Identification rate(%)	78.9	79.85
Time(s)	0.5+0.5	1+1

From these experiments, we find that for the same descriptor size (Features number: $N = n_x n_y n$ with $n = 8$ is number of bins), the best cells arrangement is always when n_x or n_y are different from 1, that is, grids of n_y horizontal stripes (e.g. 8 columns * 1 row) or n_x vertical stripes (e.g. 1 column * 8 rows) are

Table 5: Identification rate versus cells arrangements (Arrangement with and without overlapping)

Cells number	Cells arrangement	Features number	Selected features	Identification rate (%)	Time (s)
8	8*1	64	32	72.6	2+2
	4*2		37	80.4	2+2
	1*8		35	81.3	2+2
	2*4		38	**84.72**	2+3
20 without overlapping	20*1	160	57	77.45	3+4
	4*5		56	84.15	2+4
	5*4		58	**84.93**	3+4
	2*10		57	84.85	3+4
	1*20		14	76.77	4+1
20 with overlapping	20*1	160	57	72.37	3+4
	4*5		56	80.27	2+4
	5*4		58	78.33	3+4
	2*10		57	**81.59**	3+4
	1*20		14	71.23	4+1
40 without overlapping	40*1	320	73	78.8	5+6
	2*20		16	75.87	6+1
	5*8		97	**85.92**	5+8
	8*5		82	84.73	5+7
	4*10		82	85.7	5+7
	1*40		18	71.33	6+1
40 with overlapping	40*1	320	73	70.05	5+6
	2*20		16	68.29	6+1
	5*8		97	**82.74**	5+8
	8*5		82	78.23	5+7
	4*10		82	79.63	5+7
	1*40		18	64.23	6+1
80 without overlapping	80*1	640	153	72.63	14+8
	40*2		179	82.15	14+10
	2*40		183	81.27	14+12
	20*4		252	87.22	13+14
	16*5		248	**91.22**	14+14
	5*16		245	88.34	15+14
	10*8		271	89.54	15+16
	8*10		236	86.29	15+15
	1*80		150	71.28	16+9
85 without overlapping	**5*17**	680	234	**91.73**	17+14

Table 6: Identification rate versus cells arrangements (PHOG)

Cells number	85 Cells
Features number	680
Selected features	257
Identification rate(%)	95%
Time(s)	18+15

not the best choice. As it can be seen, the higher identification rates are obtained for 2 columns * 4 rows (8 cells), 5 columns * 4 rows (20 cells), 5 columns * 8 rows (40 cells) and 16 columns * 5 rows (80 cells). We also note that cells arrangements with overlapping are not advantageous for HOG since they lead to a lower identification rate compared to arrangements without overlapping. We finally remark that the greater

the number of cells increases, the more the identification rate is enhanced. These conclusions are confirmed by numerous tests. One can also see that PHOG (85 cells: 1+4+16+64 since the used PHOG is of 4 levels) achieves an identification rate of 95% which is clearly better than those obtained when considering central band or others cells arrangements with or without overlapping.

6.4   Effect of feature number

In order to evaluate the effect of features number, a 10-fold cross validation experiment was employed. In this experiment an AODEsr classifier was trained and tested with different number of features. The number of features varies depending on cells number per image (as explained in subsection: multi-cell HOGs) and a constant number of orientation bins: 8 bins.

Table 7 depicts the identification rate for the different numbers of features. A plot of the identification rate versus the number of features clearly indicates that increasing the number of features, deduced per word image, enhances the identification rate and that the best rate is achieved using PHOG compared to basic HOG with a random arrangement of cells (see Fig. 16).
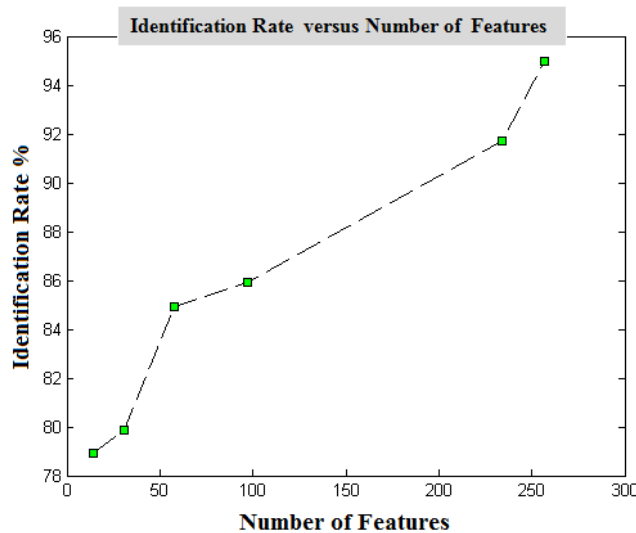

Figure 16: Identification rate versus features number.

From left to right, the first point corresponds to the identification rate (78.9%) for a features number equals to 14, obtained, when dividing the word into three bands: upper, central and lower bands, as done in Fig. 7, and selecting feature by GA (Cells number: 3 and selected feature: 14, see Table 4, the second column). The second point refers to the identification rate (79.85%) for a number of selected features equals to 31 extracted from 7 cells (2 from the upper and lower bands and 5 from cells of the central band, as shown in Fig. 8). The third point (84.93%) concerns the identification rate for a number of selected features equals to 8 which represents the best cells arrangement without overlapping from 20 cells. The fourth point (85.92%) corresponds to the identification rate for a number of selected features equals to 97 which represents the best cells arrangement with overlapping from 40 cells. The fifth point (91.73%) is the identification rate for a number of selected features equals to 234 which represents the best cells arrangement without overlapping from 85 cells. The last point (95%) refers to the identification rate for a number of selected features equals to 257 when using PHOG.

6.5   Effect of the number of orientation bins

We also tested the effect of number of orientation bins on the identification rate. The AODEsr classifier was trained with different number of orientation bins. In contrast to the cell variation, the variation of the bin size

did not show major differences in the identification rate. Fig. 17 depicts the effect of the number of orientation bins on the classification rate.
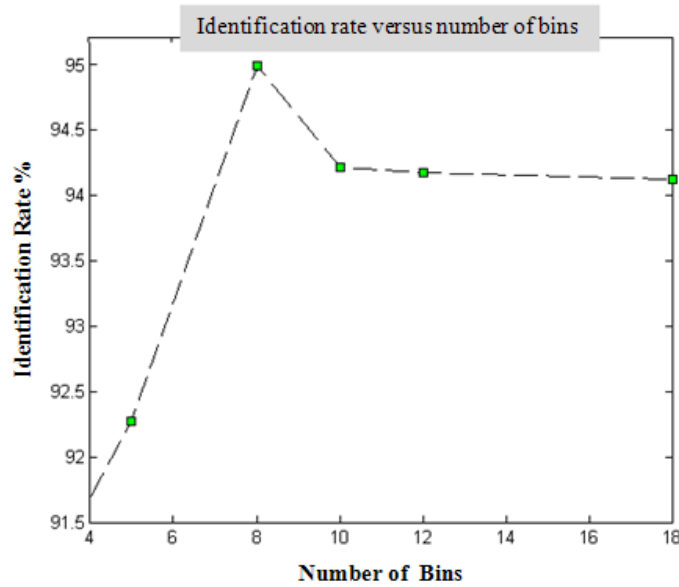

Figure 17: Identification rate versus number of orientation bins.

Table 7: Effect of orientation bins on the identification rate.

Orientation bin number	5	7	**8**	10	12	18
Features number	340	425	**680**	850	1020	1530
Selected features	154	172	**257**	331	510	728
Identification rate(%)	91.7	92.27	**95**	94.22	94.43	94.86
Time(s)	8+8	11+10	**18+15**	20+16	24+20	29+24

As it can be seen, when proposing different multi-cell HOG, with and without overlapping, then testing with PHOG, we find that the best identification rate is 95% and it is obtained using PHOG with 4 levels and 8 bins (680 features). We then suggest to combine the PHOG to co-occurrence matrices.

6.6   Effect of Co-HOG distance

As previously explained, the choice of the distance between pixels is of great importance in the computing of the Co-HOG matrices. So, we tested with different distances and chose the best one which considers any morphological change in letters. Experiments shows that the best distance is $d = 4$ (see Table 8), which is an intermediate value between the "near" and "far" distances.

Table 8: Effect of the distance on the identification rate.

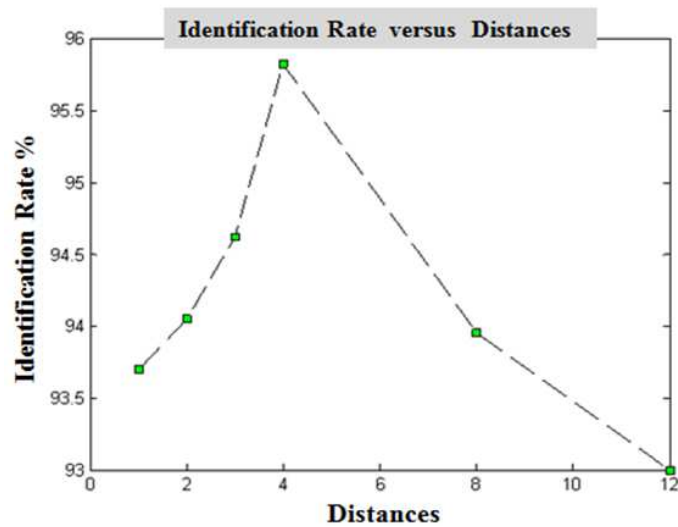| Distance	1	2	3	**4**	8	12
Features number	256	256	256	**256**	256	256
Selected features	80	73	76	**87**	96	85
Identification rate(%)	93.7	94.05	94.62	**95.82**	93.95	93

Figure 18: Identification rate using Co-HOG versus distances.

From a word image, 4 co-occurrence matrices can be extracted. The size of feature vector extracted from the Co-HOG methods is $64 * 4 = 256$.

6.7   Results from CP-HOG combination

The final vector feature is formed by a concatenation of PHOG and Co-HOG image vectors. The dimension of the final CP-HOG descriptor is equal to $680 + 256 = 936$. Like many applications, the number of features needs to be reduced due to the existence of irrelevant, noisy and redundant information which can be the detrimental elements leading to the inaccuracies in classification performance. Moreover, as the number of features used for classification increases, the number of training samples required for statistical modeling and/or learning systems grows exponentially. For this reason, we used GA which is an efficient dimensionality reduction techniques. In fact, when using GA, the number of features is reduced from 936 to 268.

6.8   Identification with different classifiers

We compared the performance of 4 typical classifiers: AODEsr, k-Nearest Neighbor (k-NN), Support Vector Machine (SVM) and Naïve Bayes (see Table 9) using two features selection methods: GA and PCA. Note that GA is 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 [20]. When applying PCA, the number of features is reduced from 936 to 274 and to 268 if GA is used.

As mentioned before, we used AODEsr classifier. Because the conditional independence is not expected to be satisfied in practical applications, naïve Bayes-like classifiers are proposed to enhance naïve classifier by relieving the restriction of conditional independence. AODEsr is a probabilistic classification learning technique which was developed to address the attribute-independence problem of the popular Bayes classifier. The 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. 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 as explained by [18]. As it can be seen AODEsr, with GA features selection method, provides good results in comparison to the others classifiers. An average identification rate of 99.07% is achieved. Note that handwritten Arabic words are correctly identified in 99.5(%) of cases. To highlight the usefulness of features selection step, we compared with a random forest classifier which proceeds without features selection and we found that it achieves an identification rate of 94.5% which is clearly inferior to 99.07%. Table 10 shows identification rate by class when using AODEsr.

Table 9: Identification rate by classifier, using different features selection methods.

	Co-HOG and GA (%)	Co-HOG and PCA (%)
SVM	95.52	95.70
k-NN	93.87	79.1
Naïve Bayes	90.60	86.1
AODEsr	99.07	96.85

Table 10: Identification rate by class.

	PA	HA	PL	HL	Average
Identification rate (%)	98.9	99.5	99.3	98.4	99.07

When observing the confusion matrix (see Table 11), we noted that it is about confusion cases between handwritten Latin and Arabic scripts which mainly come from their cursive nature. In Fig. 19(a), a handwritten Latin word has been identified as machine-printed Latin because letter stems and legs are straight. In Fig. 19(b), a handwritten Latin word has been classified as handwritten Arabic because some scribes write letters tiled to the left as generally done in handwritten Arabic script. In Fig. 19(c), a machine-printed Arabic word has been confused with handwritten Latin word due to legs which are oriented to the right.

Notice that the origin of confusion in Fig. 19(a) is caused by the lack of connections between letters. When we tested with straight words but with linked letters (see Fig. 19(d)), their script have been correctly identified. If the handwritten Latin word in Fig. 19(b) was confused with a handwritten Arabic word, it is not only and mainly caused by the right to left orientation, specific to the Arabic script, but also because the word contains few letters and these letters are disconnected. Consequently, there was not enough information on the word overall orientation to distinguish it from a handwritten Arabic word. Indeed, we tested with other handwritten Latin words whose stems are tilted to the left and these words have been properly classified (see Fig. 19(e)). This is explained by the fact that there are enough connected letters helping to correct the overall word orientation.

Table 11: Confusion matrix.

	PA	HA	PL	HL
PA	989	7	2	2
HA	1	998	0	1
PL	2	5	992	1
HL	0	9	7	984

Using CP-HOG features and after features selection (268 among 936 features) and AODEsr classifier, the proposed script identifier system captures significant amount of the differences between machine-printed and handwritten Arabic and Latin words providing a good solution for this task. In fact, the HOG of more than
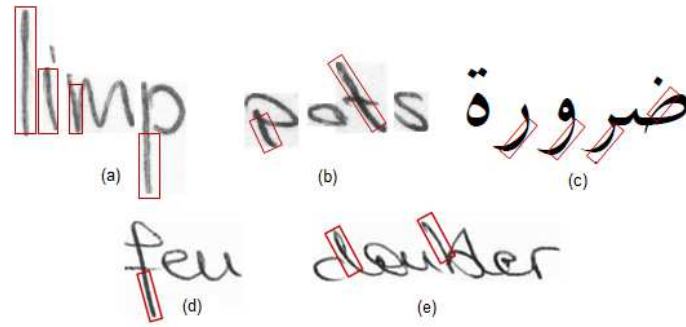
Figure 19: (a) Confusion in Latin handwritten/machine-printed identification, (b) Confusion in handwritten Latin/Arabic identification, (c) Confusion in machine-printed/handwritten and Arabic/Latin identification, (d) Correct type script identification of straight handwritten Latin word, (e) Correct script and type identification of inclined handwritten Latin word.

86.3% of Arabic handwritten words have a predominant orientation in the interval $\left[\frac{\pi}{2}, \frac{3\pi}{4}\right]$ (see Fig. 20) whereas the HOG of 84.7% of Latin handwritten words have significant spikes in the directions $\left[\pi, \frac{5\pi}{4}\right]$.
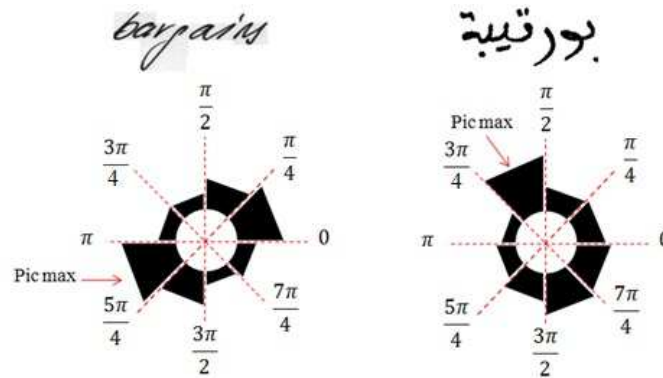

Figure 20: 8-bin HOG descriptors of Arabic and Latin words.

6.9   Comparing with existing identification system

We compared CP-HOG (our proposed system) with systems proposed by [7] and [28] which deal with the same problem. To compare CP-HOG with the steerable pyramid transform, proposed in [7] work, we used a common database (our database composed of 4000 word samples). Table 12 summarizes the obtained results.

Table 12: Comparing with existing systems.

System	Features number	Classifier	Database	Identification rate
[7]	48	k-NN	800	97.5%
[14]	48	k-NN	4000	93.32%
CP-HOG	268	AODEsr	4000	99.07%
[28]	64 per window	GMMs	20000	99.10%
CP-HOG	210	AODEsr	20000	99.90%

Notice that when testing with steerable pyramid transform [14], as done by [7] but on a larger database (4000 words instead of 800 words), the identification rate was reduced from 97.5% to 93.32%. As it can be

seen, the use of CP-HOG features with AODEsr classifier achieves an identification rate of 99.07% which is clearly better than 93.32% (the identification rate obtained using features from the steerable pyramid transform with *k*-NN as classifier and tested on the same database). We also compared our system to the one proposed by [28] on their database (20000 words). We noted a significant improvement in the identification rate (99.90%) which demonstrates the merits of the proposed system.

7   Conclusion and Prospects

In this work, we investigated the use of HOG based shape descriptors to discriminate between machine-printed and handwritten Arabic and Latin words. It is worth to note that HOG descriptors are easy to implement, characterized with high robustness against intra-class variability and fast enough for real time classification. In addition, they are scale invariant which means that words can be classified without any need for size normalization. An other merit of HOG is the robustness against deformations because slight shifts and affine deformations make small histogram value changes. Thereby the HOG descriptor has become one of the most popular low-level image representations in computer vision. Even a small improvement in its ability to represent images would be useful. In this work, we proposed to improve HOG, optimizing it for specific task of word script and type identification and we found that CP-HOG combination outperforms not only descriptors here proposed, but also those proposed in the literature. In sum, various descriptors derived from basic HOG (multi-cells HOG with and without overlapping, pyramid HOG and Co-occurrence HOG) are presented, tested and evaluated under the same experimental conditions. We also studied the influence of each feature set on performance, concluding that cells arrangements, number of features, number of orientation bins and Co-HOG distance are all important for good results. Experimental results also showed the identification process is robust and reliable at the word level. We concluded that all HOG descriptors seem to be useful to capture significant amount of the differences between Arabic and Latin scripts whether printed or handwritten. In particular, we demonstrated that a Bayes-based classifier, using a combination of Co-HOG with PHOG, can effectively solve the problem of machine-printed/handwritten and Arabic/Latin word discrimination and greatly outperform all the previous features. In fact, this combination gives near-perfect word separation and provides excellent performance relative to other existing feature sets. In the future, we will investigate some global features and combine them with CP-HOG to formulate a more accurate script identification technique. We also plan to extend the proposed system to multi-script scenarios considering other scripts written with the Arabic alphabet such as Persian, Kurdish and Urdu.s

References

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

[2] S. Mozaffari, P. Bahar, "Farsi/Arabic handwritten from machine-printed words discrimination", *Proc. of ICFHR*,Japan, pp. 694-699, (2012).

[3] E. Kavallieratou, S. Stamatatos, "Discrimination of machine-printed from handwritten Text using Simple Structural Characteristics", *Proc. of ICPR*, Cambridge, pp. 437-440, (2004). DOI: 10.1109/ICPR.2004.1334152

[4] Y. Zheng , C. Liu, X. Ding, "Single character type identification", *Proc. of DRR*, San Jos, pp. 49-56,(2002)

[5] A. M. Elgammal, M. A. Ismail, "Techniques for Language Identification for Hybrid Arabic-English Document Images", *Proc. of DRR*, Seattle, pp. 1100-1104, (2001). DOI 10.1117/12.450738

[6] K. Baâti , S. Kanoun, M. Benjlaiel, "Diffrenciation d'écriture Arabe et Latine de natures Imprimée et Manuscrite par approche globale", *Proc. of CIFED*, Tunisia, pp. 313-324, (2010).

[7] M. Benjelil , R. Mullot, M. A. Alimi, "Language and script identification based on Steerable Pyramid Features", *Proc. of ICFHR*, Bary-Italy, pp. 712-717, (2012).

[8] S. Haboubi , S. S. Maddouri, H. Amiri , "Discrimination between Arabic and Latin from bilingual documents", *Proc. of CCCA*, Tunisia, pp. 55-59, (2011).

[9] Y. Zheng , H. Li, D. Doermann, "Machine-printed text and handwritten identification in noisy documents images",*Proc. DRR*, USA, pp. 4337-353, (2004).

[10] L. Zhou , Y. Lu, C. Tan , "Bangla/English Script Identification based on Analysis of Connected Components Profiles", *Proc. of DRR*, California, pp. 243-254,(2006).

[11] M. Benjelil, S. Kanoun, A. M. Alimi, R. Mullot, "Three decision levels strategy for Arabic and Latin texts differentiation in printed and handwritten natures", *Proc. of ICDAR*, Brazil, pp. 1103-1107, (2007). DOI: 10.1109/ICDAR.2007.4377086

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

[13] H. Ma, D. Doermann , "Word level script identification for scanned document images", *Proc. of DRR(SPIE)*, San Jos, pp. 124-135, (2004).

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

[15] M. Benjleil, R. Mullot, "Performance of curvelets, dual-tree complex wavelet and discrete wavelet transform in handwritten word classification". *Proc. SoCPaR*, Tunisia, pp. 53-58, (2014). DOI: 10.1109/SOC-PAR.2014.7007981

[16] S. Bolan, L. Shijian, T. Shangxuan, H.L Joo, L.T. Chew, "Character recognition in natural scenes using convolutional co-occurrence hog", *Proc. ICPR*, Stockholm, pp. 2926-2931, (2014). DOI: 10.1109/ICPR.2014.504

[17] A. Bosch, A. Zisserman, X. Munoz, "Representing shape with a spatial pyramid kernel", *Proc. ACM International Conference on Image and Video Retreival*, Amsterdam, pp. 401-408, (2007). DOI: 10.1145/1282280.1282340

[18] C. Cortes and V. Vapnik, "Support-vector networks", *Machine Learning*, 20(3), pp. 273-297, (1995). DOI: 10.1023/A:1022627411411

[19] N. Dalal, B. Triggs, "Histograms of oriented gradients for human detection", *Vision pattern recognition, IEEE*, San Diego, pp. 886-893, (2005). DOI: 10.1109/CVPR.2005.177

[20] D.A. Forsyth, J. Ponce, "Computer vision : A modern approach", *1st edition. Prentice Hall*, pp. 1787-1797, (2002).

[21] E. Grosicki, M. Carr, J. Brodin, E. Geoffrois, "Results of the rimes evaluation campaign for handwritten mail processing", *Proc. ICDAR*, Spain, pp. 941-945, (2009). DOI: 10.1109/ICDAR.2009.224

[22] L. Jin, Y. Bai, Q. Huang, "A noval feature extraction method using pyramid histogram of orientation gradients for smile recognition", *Proc. ICIP*, Egypt, pp. 3305-3308, (2009). DOI: 10.1109/ICIP.2009.5413938

[23] A. Kacem, A. Saïdani, A. Belaïd, "How to separate between machine-printed/handwritten and arabic/latin words ?" *ELCVIA : Electronic Letters on Computer Vision and Image Analysis*, 13(1): pp. 1-16, (2014). DOI: http://dx.doi.org/10.5565/rev/elcvia.572

[24] L. Ladha, T. Deepa, "Feature selection methods and algorithms", *International Journal on Computer Science and Engineering* 3(5): pp. 1787-1797, (2011).

[25] V. Märgner, N. Ellouze, H. Amiri, M. Pechwitz,, M. S. Snoussi, "Ifn/enit - database of handwritten arabic words", *Proc. CIFED*, Tunisia, pp. 129-136, (2002).

[26] U. Marti, H. Bunke, "A full english sentence database for off-line handwriting recognition", *Proc. ICDAR*, India, pp. 705-708, (1999). DOI: 10.1109/ICDAR.1999.791885

[27] A. Mezghani, S. Kanoun, M. Khemakhem, H. El Abed, "A database for arabic handwritten text image recognition and writer identification", *Proc. ICFHR*, Italy, pp. 399-402, (2012). DOI: 10.1109/ICFHR.2012.155

[28] A. Mezghani, F. Slimane , S. Kanoun, V. Märgner, "Identification of Arabic/French-Handwritten/Printed Words using GMM-Based System", *Proc. CIFED*, France, pp. 371-374, (2014).

[29] R. Minetto, N. Thome, M. Cord, N. Leite, J. Stolfi, "T-hog : An effective gradient-based descriptor for single line text regions", *Pattern Recognition*, 46(3): pp. 1078-1090, (2004). DOI: http://dx.doi.org/10.1016/j.patcog.2012.10.009

[30] A. Saïdani, A. Kacem, "Pyramid histogram of oriented gradient for machine-printed/handwritten and arabic/latin word discrimination", *Proc. International Conference of Soft Computing and Pattern Recognition IEEE*, Tunisia, pp. 267-272, (2014). DOI: 10.1109/SOCPAR.2014.7008017

[31] T. Shangxuan, L.y. Shijian, L.T. Chew, "Scene text recognition using co-occurrence of histogram of oriented gradients", *Proc. ICDAR*, Washington, pp. 912-916, (2013). DOI: 10.1109/ICDAR.2013.186

[32] F. Slimane, R. Ingold, S. Kanoun, M.A. Alimi, "A new arabic printed text image database and evaluation protocols", *Proc. ICDAR*, Barcelona, pp. 946-950, (2009). DOI: 10.1109/ICDAR.2009.155

[33] Y. Song, B. Zhang, S. Guan, "Historic chinese architectures image retreival by svm and pyramid histogram of oriented gradients features", *International Journal of Soft Computing*, 5(2): pp. 19-28, (2010). DOI: 10.3923/ijscomp.2010.19.28

[34] I. Suthasinee, H. Punyaphol, L. Guo, Q. Huang, "Handwritten character recognition using histograms of oriented gradient features in deep learning of artificial neural network", *Proc.International Conference on IT Convergence and Security*, Macao, pp. 1-5, (2013). DOI: 10.1109/ICITCS.2013.6717840

[35] D. Thanh-Toan, E. Kijak, "Face recognition using co-occurrence histograms of oriented gradients", *Proc. ICASSP*, Japan, pp. 1301-1304, (2012). DOI: 10.1109/ICASSP.2012.6288128

[36] S. Tian, S. Lu, B. Su, C.L. Tan, "Scene Text Recognition using co-occurrence of histogram of oriented gradients", *Proc. ICDAR*, Washington, pp. 912-916, (2013). DOI: 10.1109/ICDAR.2013.186

[37] T. Watanabe, S. Ito, K. Yokoi, "Co-occurrence histograms of oriented gradients for human detection", *Transactions On Computer Vision And Applications*, 2(3): pp. 39-47 , (2010). DOI: 10.2197/ipsjtcva.2.39

[38] S. Haboubi, M. S. Snoussi, N. Ellouze, "Differenciation de documents textes arabe et latin par filtre de gabor", *Proc. TAIMA*, Tunisia, pp. 55-59, (2007).