Classification of Arabic Writing Styles in Ancient Arabic Manuscripts

Mohamed Ezz[1]
Computer and Information Sciences
Jouf University, Saudi Arabia
Systems and Computers Eng. Dept.
Al-Azhar University, Caio, Egypt

Mohamed A. Sharaf[2], Al-Amira A. Hassan[3]
Systems and Computers Eng. Dept.
Al-Azhar University
Caio, Egypt

***Abstract*—This paper proposes a novel and an effective approach to classify ancient Arabic manuscripts in "Naskh" and "Reqaa" styles. This work applies SIFT and SURF algorithms to extract the features and then uses several machine learning algorithms: Gaussian Naïve Bayes (GNB), Decision Tree (DT), Random Forest (RF) and K-Nearest Neighbor (KNN) classifiers. The contribution of this work is the introduction of synthetic features that enhance the classification performance. The training phase encompasses four training models for each style. For testing purposes, two famous books from the Islamic literature are used: 1) Al-kouakeb Al-dorya fi Sharh Saheeh Al-Bokhary; and 2) Alfaiet Ebn Malek: Mosl Al-tolab Le Quaed Al-earab. The experimental results show that the proposed algorithm yields a higher accuracy with SIFT than with SURF which could be attributed to the nature of the dataset.**

***Keywords*—Arabic manuscripts; classification; feature extraction; machine learning; GNB; DT; RF; K-NN classifiers; SURF; SIFT**

I. Introduction

Ancient manuscripts (AMs) are considered references for several centuries in history and witness on human literature and development. AMs are held in high esteem by national archives (NAs), museums and libraries all over the world. Nevertheless, NAs – in many countries – still go with traditional procedures when dealing with manuscripts. Yet, they count on experts' talents to manually process and handle manuscripts. This is considered a big concern when we deal with a heritage of thousands of years of human literature. Furthermore, to keep a manuscript in sound condition, restoration and preservation processes are applied on degraded manuscripts before pursuing any document learning procedures. For example, National Archives of Egypt (N.A.E.) has several digitization projects that work on manuscripts to achieve better results. The aim behind this work is to recognize the style of AMs. Hence, we build a model that is trained on a labeled dataset. Then we use the model to recognize a test set that has never been exposed to the model to evaluate the performance of the model before deploying it. Since Arabic manuscripts have several styles of writing which differ according to area, country, occasion and materials. They also have some characteristics as writing tools for handwritten scripts, i.e., calligraphy pens, writing direction and orientation (right to left style). In our study, we take into consideration the horizontal projection profiles of Arabic texts that have a single peak around the middle of the text-line and the alphabet letters whose shapes differ according to the location of a letter – beginning, middle or at the end of the word. Another level of difficulty is that Arabic manuscripts vary over the ages, from writer to another and this introduces variability (in the learnt features). Several books explain each style in the two modes: whole words, and separate letters. For examples, Mosoet Al Khat Al Araby comes in two volumes: (Vol. I) for Naskh [1] and (Vol. II) for Reqaa [2]. The two volumes handle each letter in the cases, i.e., words and isolated letters.

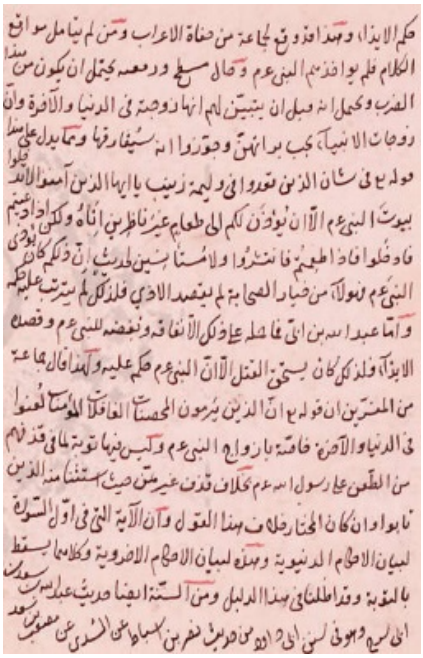
Naskh style has been derived from the "Thuluth" style and has evolved to its own form during the 10^{th} century. In addition, Naskh style is a simple and more legible style, especially in small font sizes. Also, its lines are thin and naturally round. Moreover, Naskh has become the most popular style in Arabic book publishing in general and the holy Quran in particular [3]. However, Reqaa is the simplest style for everyday non-official handwriting. It has a round fluid style [4] and was introduced in the 9^{th} century. Due its simplicity, Reqaa has become the favorite style in the eastern Arab world for everyday writing. Its words are of dense ligature structure, thick baseline and short horizontal strokes [5]. However, the scope of this work is limited to two styles: Naskh and Reqaa.

In addition, the importance of this research stems from the fact that Arabic script is the third most commonly used writing system in the world by the number of users after Latin and Chinese scripts and the second by the number of countries [6]. Also, recognizing the style of handwriting in Arabic manuscripts helps in identifying the origin and date of ancient documents and adds a step toward a large-scale digitization process. While text analysis forms the basis of object recognition and classification in several domains, our work can help in building a database for all Arabic fonts. Fig. 1 shows samples of both styles, Naskh and Reqaa.

II. Related Work

A great body of research has discussed the recognition and classification of handwriting styles in ancient manuscripts. The work of Adam, Al-Maadeed and Bouridan [7] has focused on letters that have been segmented manually from manuscripts. Then they apply Gabor Filter (GF) to extract features. The classification is accomplished by support vector machines (SVM) and yields a recognition rate close to 82% that increased to 86.84% when local binary pattern feature vector was added to GF vector. The work of Yosef et al. [8] uses

(a) Naskh style.



(b) Reqaa style.

Fig. 1. Font styles.

topological feature of letter Aleph to classify documents and the case study was the ancient Hebrew documents. Amin and Mari [9] propose a system for segmentation and recognition of characters and words that uses horizontal and vertical projections and shape-based primitives. Gillies et al. [10] split words into overlapping vertical segments. Then the location of each segment is compared against the resultant locations from the hidden Markov model (HMM). Siddiqi and Vincent [11] have applied a similar idea to solve the problem of writers identification. The method is based on the presence of the following features in a handwriting manuscript: certain patterns, orientation and curvature. For evaluation purposes, several languages are tested. However, the reported accuracy for Arabic handwritten texts was up to 92% for 100 writers. A hybrid convolutional neural network (CNN) and SVM model [12] for handwritten digit recognition is designed to automatically extract features from the raw images and yield predictions. It used non-saturating neurons and a very efficient GPU implementation of the convolution operation to reduce over fitting in the fully-connected layers. Both artificial neural network (ANN) and SVM were used in [13] to recognize Arabic numbers that have been written in different styles. A multi-agent approach to segment Arabic handwriting words [14] relies on recognition to verify the validity of the candidate segmentation points. The proposed approach uses seven agents to figure out regions where segmentation is not allowed. Tensmeyer et al. [15] present a simple CNN-based framework for classifying page images or text lines into font classes. They achieve 98.8% text line accuracy on the King Fahd University Arabic font database. Echi et al. [16] propose a set of features that have been employed successfully for the discrimination between handwritten and machine-printed Arabic and Latin scripts.

III. Proposed Model

Our training data are collected from historical books in Arabic calligraphy. Arabic by nature is a cursive language. This inspires us to introduce a novel training model that is depicted in Fig. 2. The model focuses on letters and studies all possible relative locations of a letter in inscriptions. Then we divide the combinations into four groups and build a model for each. Therefore, the final models after training: Letter at Start of inscription (LST) model which contains all characters once at the beginning of inscription, Letter at Mid of inscription (LMT) model which contains all characters once at the middle of a word, Letter at End of Work (LET) model which contains all characters once at the end of a word and Composite Training Model (CTM). CTM is the focal point of our attention because it encompasses all previous models. Hence, CTM is a model that contains every letter in the alphabet joined cursively with the set of letters in alphabet. Table I gives examples of how to construct the four models, LST, LMT, LET, and CTM.

As stated earlier, we limit our case studies to "Naskh" and "Reqaa" styles. In training stage, we study each letter with all other letters by applying the following procedure. Let i be a letter, s.t., $\forall i \in Alphabet$ do:

Build four models for letter i as follows:

1) Build a model in which each letter i pairs cursively with all letters $\in Alphabet$, i.e., CTM model,
2) Build a model in which each letter i is drawn as it comes in the beginning of a word, i.e., LST model,
3) Build a model in which each letter i is drawn as it comes in the middle, i.e., LMT model, and
4) Build a model in which each letter i is drawn as it comes at the end of a word, i.e., LET model.

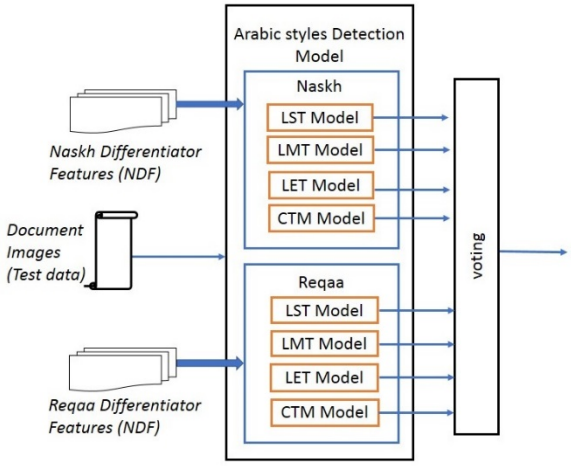


Fig. 2. Proposed classification model.

A. Training Dataset

Training dataset has been divided into eight partitions in Naskh and Reqaa styles every one represents a model. Each one of the first six models contains 24 images of 200×200 pixels, the last two contains 30 images.

B. Testing Dataset

Test dataset has two partitions in Naskh and Reqaa styles which were collected from ancient Arabic documents, historical books and Arabic calligraphy resources. The total number of images equal 200 images and are divided as follows: 1) Naskh contains 100 images of ancient documents from different ages in various fields and samples of images. 2) Reqaa contains 100 images of ancient documents, a copy of holy Quran that is written in Reqaa style and samples of images of old books that were written in Reqaa style like "Alfaiet Ebn Malek" and "Mosl Al-tolab Le Quaed Al-earb". All images in training dataset are digitized at 200×200 dpi. Then the resized images are preprocessed/filtered for noise removal. Noise in our dataset appears usually on form of isolated dots that are due to tiny drops of ink that have fallen from calligraphers' pens during the handwriting process. Several filters have been applied like mean filter, Laplacian of Gaussian filter and median filter. Empirical results show that median filter (3×3 window) yields the best results because it is highly effective in removing salt-and-pepper noise [18]. Median filter computes the median value of all the pixels under the kernel window and assign this value to the central pixel. Image set are separated into two sets: Naskh and Reqaa.

To extract the features and build our proposed approach, we need two datasets per model (LST, LMT, LE, and CTM). This mandates the construction of eight sets, four sets per style (Naskh and Reqaa). Generally, the image size is kept to 200×200 pixels.

In training phase, we kept the testing datasets (our proposed model has never trained on them) in two separate sets: First set is Naskh which contains eighty images of ancient documents from different ages in various fields and samples of images of old books were written in Naskh style like, "Al-kouakeb Al-dorya fi Sharh Saheeh Al-Bokhary". Second set is Reqaa which contains eighty images of ancient documents, a copy of the holy Quran in Reqaa style and samples of images from "classical" books in Reqaa style like, "Alfaiet Ebn Malek" and "Mosl Al-tolab Le Quaed Al-earab".

C. Feature Extraction

Khorsheed [17] proposes an approach to filter out all attributes and preserve the properties that make one character or word different from another. Features of Arabic text could be represented statistically and spatially. The statistical features analyze the spatial distribution of pixels while the structural features, the most commonly used, are based on geometrical and topological characteristic of a character, see [17] for details. Applying scale-invariant feature transform (SIFT) which uses the Difference-of-Gaussian (DOG) operator to detect distinct features in images. Lowe [18] explains that the main task of SIFT is to detect local features and describe them. While the main capacity of SIFT is preserving salient features, the big computational cost and high dimensionality of features are issues of concern. Sample runs of SURF and SIFT on the synthetic labeled dataset (CTM models for Naskh and Reqaa) are shown in Fig. 3.

Algorithm 1 learns from the labeled dataset and extracts the features. As a result, the training phase concludes with distinct features for each given class (style) of the labeled data. Therefore, the output of the training phase is either Naskh features (FN) or Reqaa features (FR).

Algorithm 1: Training Algorithm

Input: {images, labels} where labels $\in$ {'Naskh', 'Reqaa'};
Output: Features of Naskh and Reqaa styles, i.e., FN and FR;
`/* ∀ image i ∈ training set */`
for $i = 1$ *to* n **do**
  Resize i;
  Binarize i;
  Apply SIFT/SURF algorithm;
  if $i \in$ *Naskhstyle* **then**
    Save features as FN;
  else
    Save features as FR;
  end
end

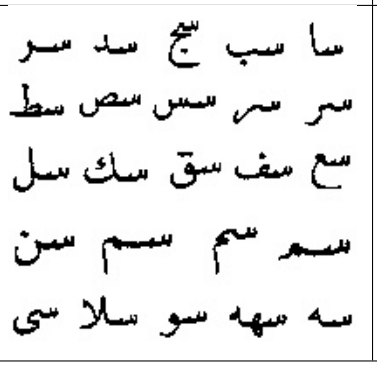
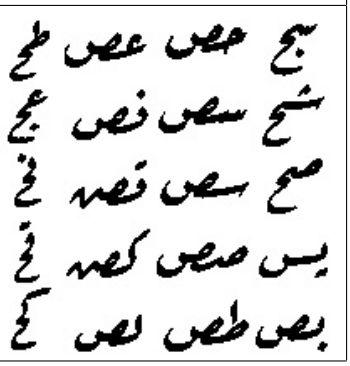
D. Image Classification

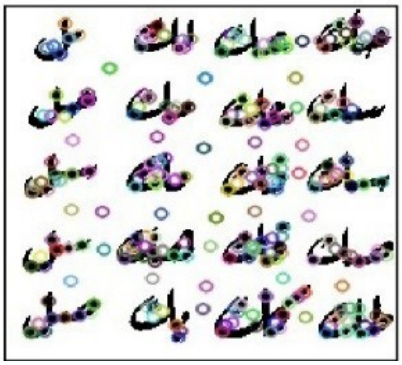
After features extraction phase, classification is conducted using the four different classifiers for performance evaluation. Algorithm 2 entails the steps that each image undergoes till a classification decision is reached. Algorithm 2 takes an image instance as an input and classify it as either Naskh or Reqaa.

IV. Experimental Results

Table II shows the results of using SIFT to learn from the four models (LST, LMT, LET and CTM) and the corresponding accuracy with the four classifiers. For the CTM

TABLE I. LST, LMT, LET, AND CTM MODELS FOR ARABIC ALPHABET

	Font Styles	
	Naskh	*Reqaa*
LST		
LMT		
LET		
CTM		



(a) Using SURF.

(b) Using SIFT.

Fig. 3. Features extraction.

Algorithm 2: Testing Algorithm

```
/* ∀ image i ∈ testing set                    */
```
Input: {image};
Output: The class to which an image belongs, i.e., $Reqaa$ or $Naskh$;
```
/* Preprocessing Stage:                       */
```
Resize image;
Binarize image;
Apply noise removal filter;
Apply SIFT / SURF algorithm and save features of image as FT;
```
/* Classification Stage:                      */
```
Apply GNB / KNN / DT / RF classifiers;
Save each classifier result;
Apply voting;
Return voting result;

TABLE II. SIFT RESULTS WITH THE FOUR CLASSIFIERS

Classifier	Style	LST	LMT	LET	CTM	Voting	Mean
GNB	NASKH	90%	91%	93%	96%	92.5%	92%
	REQAA	92%	88%	93%	95%	92%	
DT	NASKH	86%	88%	89%	89%	88%	89%
	REQAA	90%	85%	91%	94%	90%	
RF	NASKH	86%	89%	86%	93%	88.5%	88%
	REQAA	89%	86%	84%	93%	88%	
KNN	NASKH	91%	88%	90%	95%	91%	91%
	REQAA	90%	90%	91%	94%	91.3%	

model, DT classifier yields at least 89%. While RF classifier yields accuracy that is exactly 93% in both styles. In addition, KNN classifier yields accuracy that is greater than 94% in both styles. However, the best performance is achieved by GNB classifier which yields accuracy that is greater than 95%. It is apparent that the CTM model outperforms the remaining models as it is more inclusive and representative when it comes to the learnt extracted features

Similarly, the empirical performance results of GNB, DT, RF and KNN classifiers on the two case studies: Naskh and Reqaa using SURF is shown in Table III. For the CTM model, DT classifier yields accuracy that is between 87% and 89%. While RF classifier yields accuracy that is at least 90% in both styles. In addition, KNN classifier yields accuracy that is between 90% and 93%. However, the best performance is achieved by GNB classifier which yields accuracy that is greater than 91%.

Table IV shows the confusion matrices for the four training models resulting from applying GNB classifier on features extracted by using SIFT algorithm.

By analogy, Table V shows the confusion matrices attained by feeding features that are extracted by SURF algorithm and employing GNB as a classifier.

In our study, SIFT local descriptor outperforms SURF; see Fig. 3 which shows how SISFT gives higher performance than SURF.

A. Voting Procedure

Our proposed CTM contains the most distinct features of handwritten text, therefore it gives the most appealing results. As a rule of thumb in the voting process, we give CTM higher weight than the three other models. The resulting output prediction is the one that receives more than half of the votes. Table VI shows the voting process where number 1 represent class of Naskh and number 0 refers to Reqaa.

B. Performance Comparison

Finally, we conduct a performance comparison between our proposed model and two models as shown in Table VII. To have an objective comparison, a fixed setting with the literature is adopted by using the same dataset as in [15] and [19], a printed Arabic text extracted from King Fahd University Arabic Font Database (KAFD). Tensmeyer, Saunders and Martinez [15] have adopted convolutional neural network(CNN) to perform text classification. They use two sets for training: one with base-line in the manuscript and the other without a base-line. Nevertheless, their model is suffering a performance degradation when a part of the writing is cropped. While our proposed model is immune to this problem because of the plethora of features that have been introduced by the four-letter model (LST, LMT, LET, CTM). Hence, we assert that if the proposed model surpasses the existing approaches given a common dataset then we can fairly attribute this to the novel training approaches that we propose in our model.

TABLE III. SURF RESULTS WITH THE FOUR CLASSIFIERS

Classifier	Style	LST	LMT	LET	CTM	Voting	Mean
GNB	NASKH	88%	82%	81%	94%	86.3%	87%
	REQAA	89%	86%	82%	91%	87%	
DT	NASKH	84%	86%	80%	87%	84.3%	86%
	REQAA	88%	88%	85%	89%	87.5%	
RF	NASKH	85%	96%	92%	91%	91%	89%
	REQAA	86%	89%	81%	90%	86.5%	
KNN	NASKH	89%	90%	88%	90%	89.3%	90%
	REQAA	94%	91%	88%	93%	91.5%	

V. CONCLUSION AND FUTURE WORK

In this work, we propose a model for classifying Arabic writing styles in ancient Arabic manuscripts using novel models for training. To these models, we attribute the superior performance. First, the features have been extracted by using SIFT and SURF algorithms. Second, the classification stage has employed four classifiers for evaluation purposes. Then, we present the empirical performance results of GNB, DT, RF and KNN classifiers on the two case studies: Naskh and Reqaa to give results up to 92% in case of GNB classifier where KNN gives results reach to 91%. Empirically, KNN classifier gives good performance with SURF results 90%. However, the best performance in case of SIFT is achieved by GNB classifier and KNN classifier in case of SURF. Obviously, GNB and KNN classifiers have shown superior performance in Arabic manuscripts written in Naskh and Reqaa styles. Hence, the experimental results show that the learnt features from the synthetic dataset are extremely powerful in discriminating between the two styles that are considered for this study. Future work could entail conducting a comparative study of the proposed approach on other cursive languages. Also, we can introduce a meaningful weighting scheme for the voting system that guarantees an output that is at least equals to that of the CTM model.

REFERENCES

[1] K. AlGabory, *Encyclopedia of Arabic fonts, Vol.I*. Dar Alhilal, 1999.

[2] K. AlGabory, *Encyclopedia of Arabic fonts, Vol.II*. Dar Alhilal, 1999.

[3] M. Ja'Far and V. Porter, *Arabic Calligraphy : Naskh Script for Beginners*. British Museum Pubns Ltd, 2001.

[4] E. Smitshuijzen, *Arabic Font Specimen Book*. De Buitenkant, 2008.

[5] J. Janbi, "Classifying Arabic Fonts Based on Design Characteristics: PANOSE-Ao," Ph.D. dissertation, Concordia University Montreal, Quebec, Canada, 2016.

[6] N. Ghali, *Write It in Arabic: A Workbook and Step-by-Step Guide to Writing the Arabic Alphabet*, 2nd ed. Fun With Arabic, 2009.

[7] K. Adam, S. Al-Maadeed, and A. Bouridane, "Letter-based classification of Arabic scripts style in ancient Arabic manuscripts: Preliminary results," *2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*, pp. 95–98, 2017.

[8] I. B. Yosef, K. Kedem, I. Dinstein, M. Belt-Arie, and E. Engel, "Classification of Hebrew Calligraphic Handwriting Styles: Preliminary Results," *Proceedings - First International Workshop on Document Image Analysis for Libraries - DIAL 2004*, no. ii, pp. 299–305, 2004.

[9] A. Amin and J. F. Mari, "Machine recognition and correction of printed Arabic text," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 5, pp. 1300–1306, 1989.

[10] A. Gillies, E. Erl, J. Trenkle, and S. Schlosser, "Arabic Text Recognition System," *Proceedings of the Symposium on Document Image Understanding Technology*, vol. 11, no. 3, pp. 127–141, 1999. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.21.947

[11] I. Siddiqi and N. Vincent, "Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features," *Pattern Recognition*, vol. 43, no. 11, pp. 3853–3865, 2010. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2010.05.019

[12] X. X. Niu and C. Y. Suen, "A novel hybrid CNN-SVM classifier for recognizing handwritten digits," *Pattern Recognition*, vol. 45, no. 4, pp. 1318–1325, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2011.09.021

[13] L. Yang, C. Y. Suen, T. D. Bui, and P. Zhang, "Discrimination of similar handwritten numerals based on invariant curvature features," *Pattern Recognition*, vol. 38, no. 7, pp. 947–963, 2005.

[14] A. Elnagar and R. Bentrcia, "A Multi-Agent Approach to Arabic Handwritten Text Segmentation," *Journal of Intelligent Learning Systems and Applications*, vol. 04, no. 03, pp. 207–215, 2012.

TABLE IV. CONFUSION MATRICES (FOUR MODELS) FOR GNB CLASSIFIER AND SIFT

	LST		LMT		LET		CTM	
	Pred. Naskh	Pred. Reqaa	Pred. Naskh	Pred. Reqaa	Pred. Naskh	Pred. Reqaa	Pred. Naskh	Pred. Reqaa
Naskh	90	10	91	9	93	7	96	4
Reqaa	8	92	12	88	7	93	5	95

TABLE V. CONFUSION MATRICES (FOUR MODELS) FOR GNB CLASSIFIER AND SURF

	LST		LMT		LET		CTM	
	Pred. Naskh	Pred. Reqaa	Pred. Naskh	Pred. Reqaa	Pred. Naskh	Pred. Reqaa	Pred. Naskh	Pred. Reqaa
Naskh	88	12	82	18	81	19	94	6
Reqaa	11	89	14	86	18	82	9	91

TABLE VI. VOTING SCHEME

Image	LST	LMT	LET	CTM	Voting
image#1	1	0	1	1×1.5	1
image#2	0	0	1	0×1.5	0
image#3	1	1	0	0×1.5	1
...	...	...	...	...	...
image#200	0	1	0	1×1.5	1

TABLE VII. PERFORMANCE COMPARISON

Model	**Accuracy**
CNN Font Classification by Tensmeyer et al. [15]	95.05%
KAFD Arabic font database by Luqman et al. [19]	96.1%
Our Proposed Model	96.83%

[15] C. Tensmeyer, D. Saunders, and T. Martinez, "Convolutional Neural Networks for Font Classification," *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, vol. 1, pp. 985–990, 2018.

[16] A. K. Echi, A. Saïdani, and A. Belaïd, "How to separate between machine-printed/handwritten and arabic/latin words?" *Electronic Letters on Computer Vision and Image Analysis*, vol. 13, no. 1, pp. 1–16, 2014.

[17] M. Khorsheed, "Off-Line Arabic Character Recognition – A Review," *Pattern Recognition*, vol. 31, no. 5, pp. 517–530, 1998. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0031320397000848

[18] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[19] H. Luqman, S. A. Mahmoud, and S. Awaida, "KAFD Arabic font database," *Pattern Recognition*, vol. 47, no. 6, pp. 2231–2240, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320313005463