

Arabic Word Recognition System for Historical Documents using Multiscale Representation Method

Said Elaiwat¹

Department of Computer Sciences
College of Computer and Information Sciences
Jouf University
Sakakah 72441, Saudi Arabia

Marwan Abu-Zanona²

Department of Computer Sciences
College of Shari'a and Islamic Studies in Al Ahsaa
Al Imam Mohammad IbnSaud IslamicUniversity (IMSIU)
Al Ahsaa, Saudi Arabia

Abstract—In the last decades, huge efforts have been made to develop automated handwriting recognition systems. The task of recognition usually involves several complex processes including image pre-processing, segmentation, features extracting and matching. This task usually gets harder by processing historical documents as they involve skews, document degradation and structure noise. Although, the success that has been achieved in English language, the recognition of handwritten Arabic still constitutes a major challenge for many reasons. The characteristic of Arabic language, as a Semitic language, differs from other languages (e.g., European languages) in several aspects such as complex structure, implicit characters, concatenation and, writing styles and direction. This work proposes a full recognition system for the task of word recognition from Arabic historical documents. In the proposed system, a novel feature extraction method is presented to define robust features from Arabic words. Prior Feature extraction, each input image is pre-processed and segmented resulting in segmented words. After that, the features of each word/sub-word are defined based on Multiscale Convexity Concavity(MCC) analysis of contour word shape. For feature matching, a circular shift method is proposed to burn the computational cost instead of using traditional dynamic time warping (DTW) which exhibits high computational cost. Finally, the proposed algorithm has been evaluated under well-known dataset, namely, Ibn Sina, and showed high performance for historical documents with low computational cost.

Keywords—Word recognition; multiscale convexity concavity analysis; historical documents; dynamic time warping

I. INTRODUCTION

Many researchers have investigated the problem of handwritten recognition in English or Latin languages, while a few researches have targeted Arabic handwritten recognition due to the complexity of the Arabic Language. The Arabic language is an important language. More than 700 million people around the world use Arabic characters for writing and reading in Arabic or other languages like Farsi or Urdu. Arabic handwritten recognition application plays a very important role, whether online or offline, in automatic document recognition and archiving as well as in many other fields, such as office automation, cheque verification, mail sorting, and large variety of banking business [1], [2], [3]. The Arabic language, as a Semitic language, has many unique features and characteristics, which pose additional challenges for those who are interested in automatic recognition of Arabic documents.

This work proposes an efficient word recognition system for Arabic historical documents based on Multiscale Convex-

ity Concavity (MCC) representation. The proposed system was divided into five main phases, namely morphological image processing, segmentation, feature extraction, and feature matching and dissimilarity. The first phase aims to pre-processed the input image by removing noise and connecting the disconnected parts of the writing lines. The next phase (segmentation phase) involves line and word segmentation. The line segmentation method is based on defining the separable lines between text lines. Unlike other methods which rely on defining the base line, our method basically works by detecting high and low activities along with each line using image histogram. Based on these activities, the separable line between text lines is defined. After line segmentation, each shape (word or part of word) is defined based on 8 neighbour connected pixels, followed by merging one or more shapes to produce a single word.

In feature extraction phase, the contour shape is extracted and analyzed using Multiscale Convexity Concavity (MCC) followed by Discrete Cosine Transform (DCT) resulting in what is named as MCC-DCT features. Once the MCC-DCT features are obtained, the similarity between two different words is carried out by comparing (matching) their MCC-DCT features. For matching, we propose a circular shift method instead of using dynamic time warping, and thus significantly reducing the computational cost. However, this could decrease the recognition rate.

The remainder of this paper is organized as follows: Section 2 presents a summary of related work. Section 3 presents the proposed Holistic word recognition system in details. Finally, the experimental evaluation of the proposed system is presented in Section 4.

II. RELATED WORK

Feature extraction is considered on of the most important phases of word recognition and plays a crucial role to achieve robust and accurate recognition. Many statistical and structural features [4], [1], [5], [6], [3] have been widely used in word recognition along with different classifiers including hidden Markov model (HMM) [7], [8], neural network [9], [10], [11], support vector machine (SVM) [12], [13] and others. Statistical features can be represented as numerical measures defined over the whole image or some regions of the image. Several measures, including pixel densities, histograms of chain code directions and Fourier descriptors, can be used in this type

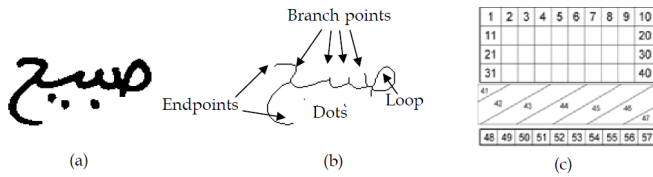


Fig. 1. An example of different type of features. (a) Original image. (b) Structural features. (c) Statistical features. [1], [14]

of features. In contrast, structural features can be represented as a composition of structural units such as loops, ascenders, descenders, branch-points and dots. These units are usually (not always) defined from the skeleton or the contour of the image. Fig. 1 [1], [14] shows an example of both statistical and structural features .

Tomasz and Noel [15] proposed a Contour-based Shape Representation for computing the similarity between two shapes, represented by 2D closed contours. This representation showed to be invariant under translation, scaling, and rotation processes. It was also found that the proposed representation could be used for database recovery or for detecting the regions with a specific shape in a sequence of images in video. El-Hajj et al [4] proposed analytical approach to define features based on baseline position. A set of measures, including densities, concavity and transitions were extracted from a patch related to the baseline. Baseline dependent features were then formulated from these measures and passed to HMM classifier for classification. Chen et al. [5] applied a set of Gabor filters to extract features (Gabor features) from Arabic handwritten words. SVM classifier was then used for classification task. In addition, they combined Gabor and gradient-structural-concavity (GSC) features to achieve better recognition rate In [16], a learned feature model was introduced for offline Handwriting Recognition by applying a statistical bag-of-features model. This model is then integrated with Hidden Markov Model (HMM) for the task of word recognition. Al-mazán et al. [17] proposed an approach for word spotting and recognition based on joint embedding space which is defined from word images and text strings. The proposed model was shown to achieve high accuracy in both Document Images and Natural Images with minimal training data. Nemouchi et al. [18] introduced an Arabic handwriting recognition system by combining methods of decision fusion approach using the HMM-Toolkit (HTK). Prior feature extraction, each image was pre-processed and segmented into lines. Sliding window technique was then used for feature extraction while HMMs was applied for classification.

More sophisticated features were introduced in [18] by combing both statistical and structural features using three different sets of features. The first set includes Zernike moments and the structural features of the word, extracted from the binary image. The second set includes Freeman code, extracted from the contour image of the word. The final set includes zoning features, extracted from the skeleton image. For classification, four classifiers, namely Fuzzy C Means algorithm (FCM), K-Means algorithm, K Nearest Neighbor algorithm (KNN) and Probabilistic Neural Network (PNN) were applied. The decision of all classifier were then combined through the simple vote and the weighted sum methods.

Chherawala and Cheriet [19] proposed an Arabic word descriptor (AWD) for the task of lexicon reduction. Their algorithm consists of two stages. The first stage computes the structural descriptor (SD) for each connected component (CC) of the word image. The second stage normalizes the structural descriptors (SDs) to form the Arabic word descriptor (AWD). The reduced lexicon was obtained by first ranking the original lexicon based on the distances between AWD of input word and AWDs of original lexicon. Those words within n top ranks in the original lexicon were used to formulate the reduced lexicon. For recognition task, the proposed lexicon reduction method was combined with different type of recognition techniques including analytic word recognition and holistic word recognition.

In the recent years, several organizations have tried to digitize a large amounts of historical handwritten documents as they could be destroyed because of their age [20]. This has led to open a new trend of text recognition, focusing on understanding and analyzing historical documents. Unlike standard handwritten documents, historical documents are characterized by low quality and large variations of writing styles making them hard to understand even by a human [21]. Different approaches have been proposed in this context including Gradient, Structural and Convexity (GSC) features [22], [23], [24], Gabor features [25], Dynamic Time Warping (DTW) [26], HOG Features [27], Multiscale Convexity Concavity features [28] and others.

III. HOLISTIC WORD RECOGNITION

The key of the proposed work is to analyze the closed contour for each world / sub word through multi-scale representation. This requires for each document to be pre-processed, segmented (line and word segmentation) prior extracting contour points.

Fig. 2 shows a full flowchart of the proposed system, starting from input image and ending up with classified word . We can notice that the proposed system is divided into four main phases, namely Morphological image processing, segmentation, Multiscale Convexity Concavity (MCC) representation, and matching and dissimilarity.

A. Morphological Image Processing

In this phase, the colored image is first converted to binary image as shown in Fig. 3.

The value of each pixel in the binary image is then normalized using morphological erosion and dilation. Morphological dilation adds pixels to the boundaries of image shape, while erosion removes pixels from image shape's boundaries. Fig. 4 shows the basic concept behind of dilation operation.

The main idea of applying Morphological erosion and dilation, here, is to connect the disconnected parts of the writing lines. As the ink intensity is not constant over all lines of the word, auto binarization could results in disconnect of the continuous lines. Dilation has the effect of merging disconnected parts of the same word, while erosion refines the word so it dose not over think.

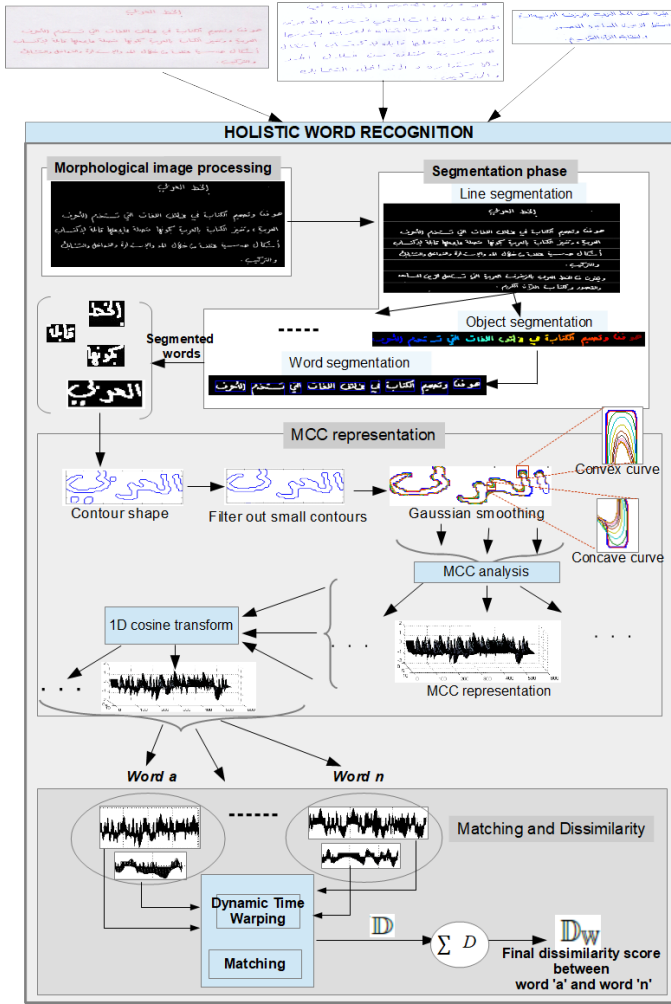


Fig. 2. Flow Chart of Proposed System

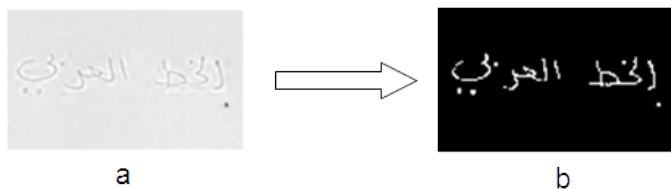


Fig. 3. a. An image before applying binarization b. An image after applying binarization

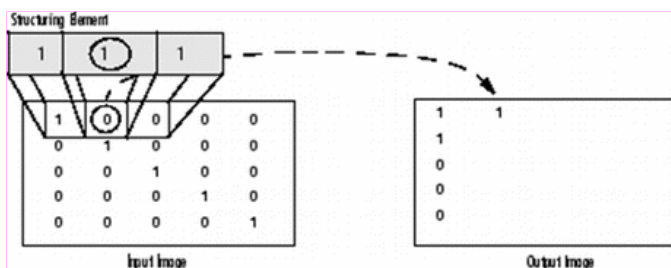


Fig. 4. Dilation Operation



Fig. 5. Define High and Low Activity Sets based on Histogram of Image Lines (Horizontal Lines)



Fig. 6. Shape Segmentation based on 8 Neighbour Connected Pixels (Colors are only used to Illustrate Different Objects)

B. Segmentation Phase

Segmentation phase is applied to segment each word in the text image. This phase consists of two parts: line and word segmentation.

1) *Line Segmentation*: The line segmentation method is based on defining the separable lines between text lines. Unlike other methods which rely on defining the base line, this method basically works by detecting high and low activity along each line using image histogram. Based on these activities, the separable line between text lines is defined. This can be achieved by defining sets of low and high activities based on threshold β . If the activity of the current line is less than β , then it belongs to low activity sets, otherwise it belongs to high activity sets. Now, each low activity set, falling between two high activity sets, contains a single separable line. This line is chosen to be the lowest line activity in the low activity set. Fig. 5 illustrates the process of line segmentation.

2) *Word Segmentation*: Prior segmenting each word, shapes at each line is segmented based on 8 neighbour connected pixels. Fig. 6 shows shape segmentation where each shape (word/sub-word) represented in different color (colors are used for illustration only). This segmentation is followed by merging multiple shapes (one or more) to perform word segmentation, producing segmented words. This merging is done according to distances between shapes. If the distance between shapes (two or more) are less than the distance threshold α , these shapes are merged to produce a single word.

By applying all segmentation phase on full text image, each word is segmented as shows in Fig. 7.

It is clear from Fig. 7 that some words are incorrectly segmented. This is because the writing style here is inconsis-

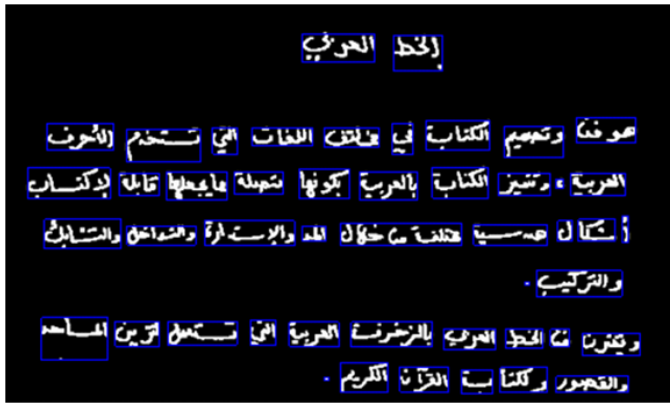


Fig. 7. Segmented Words

tent. The distance between words sometimes is less than the distance between shapes (sub-words) within the same word. This problem can be solved by adding some constraints on the writing style of writers to be consistent. Note that the value of the distance threshold was defined experimentally through a set of empirical tests. Although this way is simple and does not require any extra process, it could be inefficient in some cases, notably when the difference between writing styles is big. This problem can be minimized by investigating an automated way to define the distance threshold. For example, defining the distance threshold by analysing the minimum and the maximum distances between shapes (sub-words) for a document within the same writing style. However, this requires a deep investigation to compromise between simplicity and accuracy which can be addressed in the future work.

C. MCC Representation

After segmentation phase, closed contour shape of each word is extracted as illustrated in Fig. 8. Note that some words have multi-closed contour regarding to the number of sub-words. Multi-scale analysis, in such cases, cannot be applied directly since it is applicable only on single contour (single contour per word).

To avoid this issue, we either connect sub-words together (within a single word) prior extracting the contour shape, or extracting MCC representation from each sub-word separately (within a single word). The first approach was used by [28][29] for English word recognition. However, unlike English word, separated sub-words which represent single word, cannot be connected together as this could totally change the word. In contrast, second approach dose not change the shape of Arabic words making it suitable to represent word features. In both approaches, very small contour shapes such as dots and diacritics (e.g. “” and “.”) are ignored because they are not stable shapes, producing unstable results. Further more, most of the old historical documents do not use movements and dots in their writings. To achieve that, the size of each closed contour within a single word is evaluated using a threshold value ρ . Any contour shape, with size less than ρ , is filtered out as illustrated in Fig. 8.

Now, each contour need to be normalized to have the same number of contour points for different shapes. It is worth

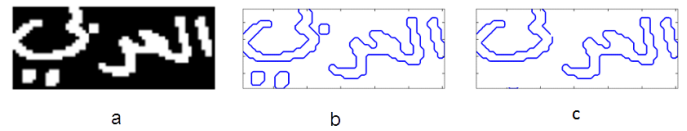


Fig. 8. a. Cropped Word, b. Contour Shape Extraction, c. Filtering Out Small Contours

Arabic sub-words	Contour shape	# of contour points
رجلسح	رجلسح	1312 points
سح	سح	743 points
ه	ه	340 points

Fig. 9. An Example of Arabic Sub-Words with Varied Length

noting that number of contour points is varied regarding to the size of the shape (sub-word) and no limitation on that. An example of such cases illustrated in Fig. 9.

Different strategies can be applied here in order to unite the length of all contour shapes. It is good to define a selection criteria to extract a fixed number of dominant points from each contour. However, a such criteria requires a lot of investigation before applying. Instead, each contour shape is sampled to have N unit length (points).

After length normalization, each contour point is analysed using Gaussian kernel. Since each contour point includes the value of two dimensions (x and y), Gaussian kernel is applied for each dimension separately. Suppose we have a contour C with N points represented by $(x(u), y(u))$ coordinators, where $u \in \{1, 2, \dots, N\}$. The coordinators, $x(u)$ and $y(u)$ are convolved with Gaussian kernel ψ_σ resulting a smoothed contour C_σ at scale σ with x_σ and y_σ coordinators as the following,

$$\begin{aligned} x_\sigma(u) &= \int x(u)\psi_\sigma(\tau - u)dt \\ y_\sigma(u) &= \int y(u)\psi_\sigma(\tau - u)dt \end{aligned} \quad (1)$$

$$\psi_\sigma(\tau) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{\tau^2}{2\sigma^2}}$$

the kernel size is set to be fixed for all scales. Note that the value of scale σ determines the smoothness degree of the Gaussian kernel ψ_σ . For simplicity, we assume that σ takes an integer values between 1 to 10.

To define the convex and concave peaks, each contour point is evaluated based on the difference (displacement) occurs through consecutive scales. More specifically, The displacement of each contour point through consecutive scales are used to build a rich multiscale representation of convexity and concavity. Assume that a contour point u at scale σ denoted as $(x_\sigma(u), y_\sigma(u))$, (as shown in Eq. 1). The MCC representation

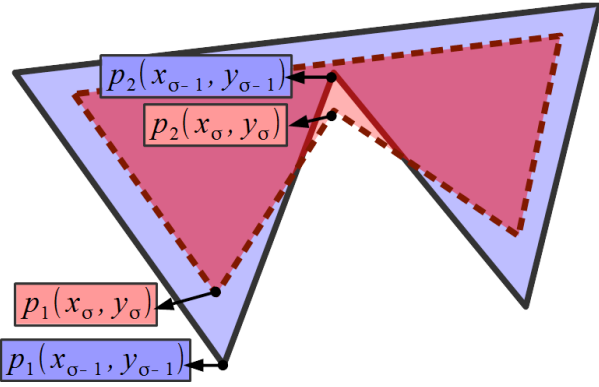


Fig. 10. An Example of Convex and Concave Peaks

$M(\sigma, u)$ is defined as follows,

$$M(\sigma, u) = \nu \sqrt{(x_\sigma(u) - x_{\sigma-1}(u))^2 + (y_\sigma(u) - y_{\sigma-1}(u))^2}$$

$$\text{where, } \nu = \begin{cases} 1, & \text{if } (x_\sigma(u), y_\sigma(u)) \text{ inside } C_{\sigma-1} \\ -1, & \text{Otherwise} \end{cases} \quad (2)$$

We can note that $M(u, \sigma)$ represents the displacement in contour point u through scale σ and scale $\sigma-1$. Contour points with sharp convexity/concavity produce large displacements over different scales. The variable ν in Eq. 2 was used to distinguish between convex and concave curves. A positive value was assigned to $M(u, \sigma)$ if it represents a convex curve. In contrast, a negative value was assigned to $M(u, \sigma)$ if it represents a concave curve. Fig. 10 shows displacements in the both convex and concave curves of contour C through scales $\sigma-1$ and σ . Point p_1 of the inner contour (C_σ) falls inside contour $C_{\sigma-1}$ leading to convex peak (positive value). In contrast, Point p_2 of contour C_σ falls outside contour $C_{\sigma-1}$ leading to concave peak (negative value).

Fig. 11 shows an example of MCC extraction. Note that each element inside MCC matrix has been represented by three values namely, position u , scale σ and displacement d .

MCC-DCT: Further improvement can be applied on MCC representation by reducing the correlation between multi-scale components (σ, s) for each contour point. This can be done by applying 1D Discrete Cosine Transform (DCT) on each column of MCC matrix. Now, Instead of using displacement d as the case of MCC, decorrelated elements (coefficients) resulting from DCT are used to build MCC-DCT \hat{M} as the following,

$$\hat{M}(\varphi, u) = \sqrt{2/L} s(k) \sum_{\sigma=1}^L M(\sigma, u) \cos \left[\frac{\pi}{L} (\sigma + 0.5) k \right]$$

$$\text{where, } s(k) = \begin{cases} \sqrt{0.5}, & \text{if } (k = 0) \\ 1, & \text{otherwise} \end{cases} \quad (3)$$

where L represents the number of scales while s represents a scaling function.

D. Matching and Dissimilarity

In this section, a contour matching algorithm was first proposed in order to measure the dissimilarity between MCC

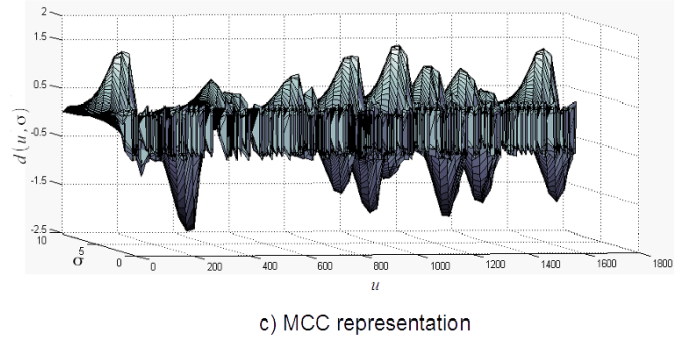
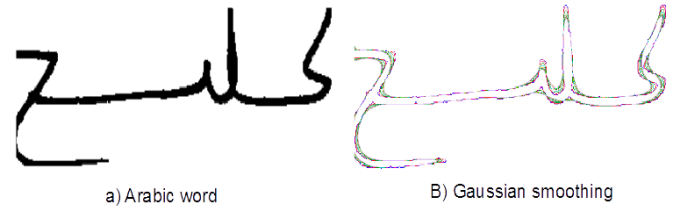


Fig. 11. An Example of MCC Representation

matrices. Since each word may have more than single contour, a Word-to-word matching strategy has then been introduced to define the the dissimilarity between two words.

1) MCC matching: The distance between two contours (MCC matrices) was carried out by matching contour points through different scales. Assume that $d(A_a, B_b)$ represents the distance between contour points $a^A \in A$ and $b^B \in B$ through scales $\sigma \in (1, 2, \dots, I)$. The distance can be defined as follows,

$$d(a^A, b^B) = \sum_{\sigma=1}^I \frac{|a_\sigma^A - b_\sigma^B|}{v_\sigma^A + v_\sigma^B} \quad (4)$$

the operators v_σ^A and v_σ^B represent a normalizing factor for each scale σ . The first operator is defined as $v_\sigma^A = \max(A_{\sigma,a})_{a=1}^N + \min(A_{\sigma,a})_{a=1}^N$ and, in the same manner, the second one is defined.

By Matching each point in A and B , we get a distance matrix D where the index (i, j) represents the distance between points a_i and b_j as shown in Fig. 12. The final dissimilarity measure between A and B is calculated by finding the cumulative distance $\mathbb{D} = \sum D_{min}/H$. D_{min} represents the optimal path (minimal diagonal distance) which obtained by using the Dynamic Time Warping (DTW) [29]. From Fig. 12, the examined indices were pointed by a blue color, while the selected path was pointed by black lines.

Matching Optimization: The complexity of defining D is $H \times H$ since we match all contour points together. Instead, we can match only corresponding points from A and B together which only cost H . To achieve that, A and B need to be reordered (reorder the columns) so they have the same sequential order based on a reference index. This index was chosen to be the highest convexity displacement between the coarsest and the finest scales in Gaussian representation. In other word, the highest convexity value resulted from summing MCC representation through different scales (rows in MCC matrix). Assume that g and q are reference indices for A and

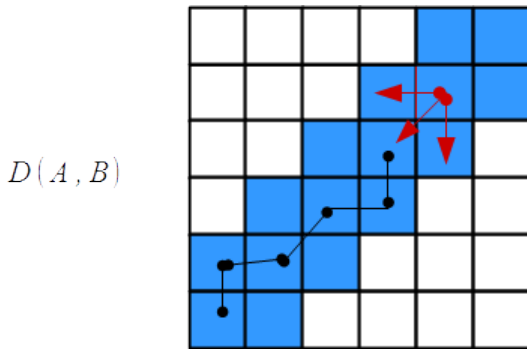


Fig. 12. Distance Matrix D and the Selection of Optimal Path

B , respectively. The reordered representations \tilde{A} and \tilde{B} can be defined as follows:

$$\begin{aligned}\tilde{A} &= \text{CircularShift}(A)_g \\ \tilde{B} &= \text{CircularShift}(B)_q\end{aligned}\quad (5)$$

where $\text{CircularShift}(x)_i$ function is shifting circularly the elements (columns) of x regarding to the reference i . Now, by matching corresponding points between \tilde{A} and \tilde{B} using Eq. 4, the distance matrix can be represented as $D = \{d(a_1, b_1), d(a_2, b_2), \dots, d(a_H, b_H)\}$

2) *Word-to-word matching*: In the previous section, we calculated the dissimilarity matrix between contour shapes. However, Arabic words usually consist of multiple shapes (sub-words). This requires to match between corresponding sub-words A_i and B_j within words W_A and W_B , respectively. The final dissimilarity measure between W_A and W_B is defined as follows:

$$\mathbb{D}_w(W_A, W_B) = \sum_{i=1}^N (\mathbb{D}(A_i, B_i))$$

where $W_A = \{A_1, A_2, \dots, A_N\}$, $W_B = \{B_1, B_2, \dots, B_N\}$ (6)

IV. EXPERIMENTAL RESULTS

In this section, the performance of the proposed system has been evaluated for the task of word recognition from historical documents. To achieve this, our system has been evaluated on well-known historical Arabic datasets, namely, Ibn Sina [30]. The name of the dataset is related to a famous Persian scholar, Ibn Sina, since it is derived from his philosophical work. This dataset consists of 60 pages with approximately 25,000 sub-words. Around 1200 different classes are represented in the dataset, distributed on different classes. Fig. 13 shows a page sample selected from the dataset.

We randomly selected images from dataset to build two sets of reference and test sets, with the ratio of 10 to 1, respectively. The reference set forms our lexicon while the test set is used to evaluate the performance of the proposed system.

Fig. 14 reports the recognition results of the proposed method on Ibn Sina dataset as a function of different ranks. The performance of both MCC-DCT and MCC-DCT-Circular shift methods are relatively close together. MCC-DCT method

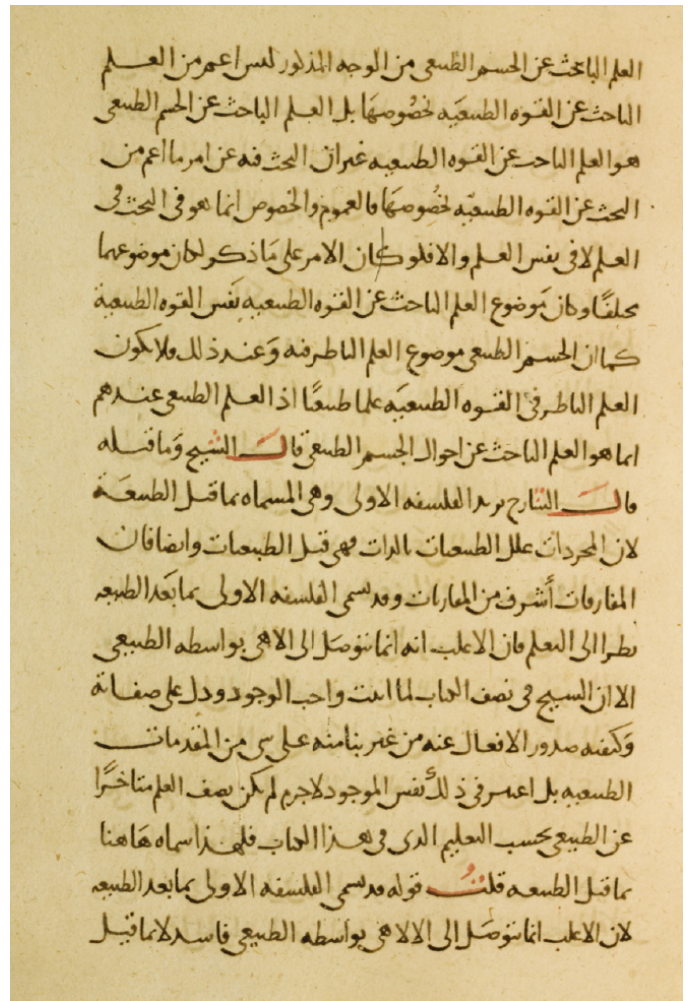


Fig. 13. A Sample from Ibn Sina Dataset

TABLE I. THE RECOGNITION RATE OF THE PROPOSED METHODS ON IBN SINA DATASET ALONG WITH THE AVERAGE PROCESSING TIME OF EACH WORD

	Ibn Sina dataset	
	Recognition rate	Avg. processing time (s)
MCC-DCT	96.6%	43.8
MCC-DCT _{circular-shift}	91.5%	0.74

achieved recognition rates of 85% at rank 1, while MCC-DCT_{circular-shift} achieved recognition rates of 84.5% at the same rank. We can note that the recognition rate of MCC-DCT_{circular-shift} is 5% less than standard MCC-DCT. However, it is a trade off between accuracy and complexity, where MCC-DCT_{circular-shift} exhibited much lower complexity than MCC-DCT. Table I shows the recognition rate of both methods along with average processing time of each word. It is clear that MCC-DCT_{circular-shift} is 59 times faster than the standard MCC-DCT.

V. CONCLUSION

In this work, we introduced a word recognition system capable to process and recognize holistic words in Arabic historical documents. The main contribution of the proposed

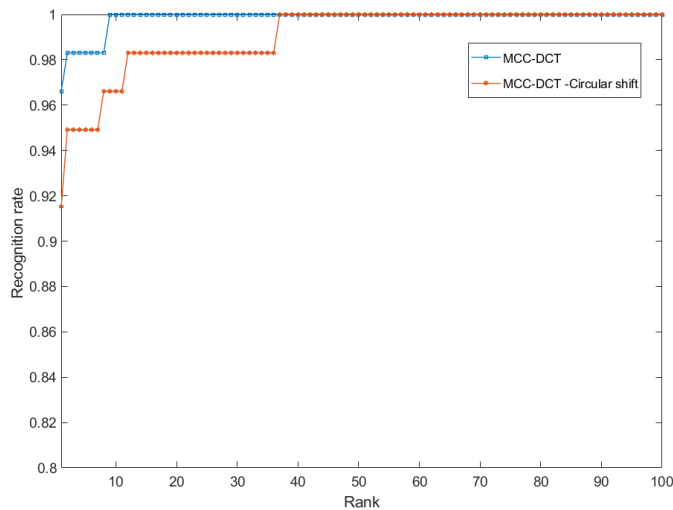


Fig. 14. Recognition Rate of our Proposed System at Different Ranks on Ibn Sina Dataset

system is to define robust features, extracted from holistic word, based on Multiscale Convexity Concavity (MCC) representation. As Arabic word could include multiple shapes, the features of each shape, within a single shape, are defined separately. For matching, the corresponding shapes between examined words are firstly matched and then the resulted distances (from matching) are combined to compute the the over all distance between these words. To avoid the high computational cost resulting from dynamic time warping match, we proposed a circular shift match which significantly burn the computational cost. Our experiments showed that our system capable to compromise between high recognition rate and low computational cost. In the future work, we plan to optimize our circular shift matching by investigating more advanced criteria to define the reference index such the energy function.

REFERENCES

- [1] L. M. Lorigo and V. Govindaraju, "Offline arabic handwriting recognition: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 712–724, May 2006.
- [2] Y. Akbari, M. J. Jalili, J. Sadri, K. Nouri, I. Siddiqi, and C. Djeddi, "A novel database for automatic processing of persian handwritten bank checks," *Pattern Recognition*, vol. 74, pp. 253 – 265, 2018.
- [3] J. A. Sánchez, V. Romero, A. H. Toselli, M. Villegas, and E. Vidal, "A set of benchmarks for handwritten text recognition on historical documents," *Pattern Recognition*, vol. 94, pp. 122 – 134, 2019.
- [4] R. El-Hajj, L. Likforman-Sulem, and C. Mokbel, "Arabic handwriting recognition using baseline dependant features and hidden markov modeling," in *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, pp. 893–897 Vol. 2, Aug 2005.
- [5] "Gabor features for offline arabic handwriting recognition," in *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, DAS 10, pp. 53–58, ACM, 2010.
- [6] M. T. Parvez and S. A. Mahmoud, "Arabic handwriting recognition using structural and syntactic pattern attributes," *Pattern Recognition*, vol. 46, no. 1, pp. 141–154, 2013.
- [7] A. El-Yacoubi, M. Gilloux, R. Sabourin, and C. Y. Suen, "An hmm-based approach for off-line unconstrained handwritten word modeling and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, pp. 752–760, Aug 1999.
- [8] A. Benouareth, A. Ennaji, and M. Sellami, "Arabic handwritten word recognition using hmms with explicit state duration," *Eurasip journal on advances in signal processing*, vol. 2008, no. 1, pp. 1–13, 2007.

- [9] J. H. AlKhateeb, J. Ren, J. Jiang, S. S. Ipson, and H. El Abed, "Word-based handwritten arabic scripts recognition using dct features and neural network classifier," in *2008 5th International Multi-Conference on Systems, Signals and Devices*, pp. 1–5, July 2008.
- [10] V. Frinken, A. Fischer, R. Manmatha, and H. Bunke, "A novel word spotting method based on recurrent neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 211–224, Feb 2012.
- [11] R. Ptucha, F. P. Such, S. Pillai, F. Brockler, V. Singh, and P. Hutkowski, "Intelligent character recognition using fully convolutional neural networks," *Pattern Recognition*, vol. 88, pp. 604 – 613, 2019.
- [12] A. R. Ahmad, C. Viard-Gaudin, and M. Khalid, "Lexicon-based word recognition using support vector machine and hidden markov model," in *2009 10th International Conference on Document Analysis and Recognition*, pp. 161–165, July 2009.
- [13] M. W. Sagheer, C. L. He, N. Nobile, and C. Y. Suen, "Holistic urdu handwritten word recognition using support vector machine," in *2010 20th International Conference on Pattern Recognition*, pp. 1900–1903, Aug 2010.
- [14] N. Farah, T. Khadir, and M. Sellami, "Artificial neural network fusion: Application to arabic words recognition.," in *ESANN*, pp. 151–156, 2005.
- [15] T. Adamek and N. O'Connor, "Efficient contour-based shape representation and matching," in *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*, pp. 138–143, ACM, 2003.
- [16] L. Rothacker, S. Vajda, and G. A. Fink, "Bag-of-features representations for offline handwriting recognition applied to arabic script," in *2012 International Conference on Frontiers in Handwriting Recognition*, pp. 149–154, Sep. 2012.
- [17] J. Almazan, A. Gordo, A. Fornes, and E. Valveny, "Word spotting and recognition with embedded attributes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, pp. 2552–2566, Dec 2014.
- [18] S. Nemouchi, L. S. Meslati, and N. Farah, "Classifiers combination for arabic words recognition: application to handwritten algerian city names," in *International Conference on Image and Signal Processing*, pp. 562–570, Springer, 2012.
- [19] Y. Chherawala and M. Cheriet, "Arabic word descriptor for handwritten word indexing and lexicon reduction," *Pattern Recognition*, vol. 47, no. 10, pp. 3477–3486, 2014.
- [20] R. F. Moghaddam and M. Cheriet, "Application of multi-level classifiers and clustering for automatic word spotting in historical document images," in *2009 10th International Conference on Document Analysis and Recognition*, pp. 511–515, July 2009.
- [21] S. Dey, A. Nicolaou, J. Lladós, and U. Pal, "Local binary pattern for word spotting in handwritten historical document," in *Structural, Syntactic, and Statistical Pattern Recognition (A. Robles-Kelly, M. Loog, B. Biggio, F. Escolano, and R. Wilson, eds.)*, (Cham), pp. 574–583, Springer International Publishing, 2016.
- [22] B. Zhang, S. N. Srihari, and C. Huang, "Word image retrieval using binary features," in *Document Recognition and Retrieval XI (E. H. B. Smith, J. Hu, and J. Allan, eds.)*, vol. 5296, pp. 45 – 53, International Society for Optics and Photonics, SPIE, 2003.
- [23] S. N. Srihari, H. Srinivasan, P. Babu, and C. Bhole, "Spotting words in handwritten arabic documents," in *Electronic Imaging*, 2006.
- [24] A. Bhardwaj, D. Jose, and V. Govindaraju, "Script independent word spotting in multilingual documents," in *Proceedings of the 2nd workshop on Cross Lingual Information Access (CLIA) Addressing the Information Need of Multilingual Societies*, 2008.
- [25] H. Cao and V. Govindaraju, *Template-Free Word Spotting in Low-Quality Manuscripts*, pp. 135–139. 2006.
- [26] T. M. Rath and R. Manmatha, "Word image matching using dynamic time warping," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, vol. 2, pp. II–II, June 2003.
- [27] K. Terasawa and Y. Tanaka, "Slit style hog feature for document image word spotting," in *2009 10th International Conference on Document Analysis and Recognition*, pp. 116–120, July 2009.

- [28] T. Adamek, N. E. OConnor, and A. F. Smeaton, "Word matching using single closed contours for indexing handwritten historical documents," *International Journal of Document Analysis and Recognition (IJ DAR)*, vol. 9, no. 2, pp. 153–165, 2007.
- [29] T. Adamek and N. E. OConnor, "A multiscale representation method for nonrigid shapes with a single closed contour," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, pp. 742–753, May 2004.
- [30] R. Farrahi Moghaddam, M. Cheriet, M. M. Adankon, K. Filonenko, and R. Wisnovsky, "Ibn sina: A database for research on processing and understanding of arabic manuscripts images," in *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, DAS '10*, (New York, NY, USA), p. 11–18, Association for Computing Machinery, 2010.