



ORIGINAL

A Progressive Approach to Arabic Character Recognition Using a Modified Freeman Chain Code Algorithm

Enfoque progresivo del reconocimiento de caracteres árabes mediante un algoritmo de código en cadena de Freeman modificado

Mohamed Rida Fethi¹ , Othmane Farhaoui¹  , Imad Zeroual¹ , Ahmad El Allaoui¹ 

¹L-STI, T-IDMS, FST Errachidia, Moulay Ismail University of Meknes. Morocco.

Cite as: Rida Fethi M, Farhaoui O, Zeroual I, El Allaoui A. A Progressive Approach to Arabic Character Recognition Using a Modified Freeman Chain Code Algorithm. Data and Metadata. 2023;2:178. <https://doi.org/10.56294/dm2023178>


Submitted: 01-03-2023

Revised: 27-03-2023

Accepted: 07-05-2023

Published: 08-05-2023

Editor: Prof. Dr. Javier González Argote 

Guest Editor: Yousef Farhaoui 

Note: paper presented at the International Conference on Artificial Intelligence and Smart Environments (ICAISE'2023).

ABSTRACT

Arabic character identification presents a significant obstacle to the comprehension and analysis of Arabic text. This paper presents an improved technique that generates Freeman code from handwritten Arabic characters. This code provides the shortest code length without losing character information, accounting for all handwritten Arabic character variants. We tested this code using a set of Arabic characters in various formats to identify Arabic characters in order to take use of the code generated by our enhanced method. We also performed a comparison between our Freeman code and codes generated in other related research. In light of this, the code that we obtained correctly represents the Arabic letter in all of its variants, including the ones that the algorithms in previous publications did not consider. Consequently, our novel method based on Freeman coding represents a significant advancement in Arabic character recognition. Furthermore, our method provides a successful way of identifying and presenting Arabic characters.

Keywords: Freeman Chain Code; Handwritten Arabic Characters; Normalization Code.

RESUMEN

La identificación de caracteres árabes supone un obstáculo importante para la comprensión y el análisis del texto árabe. Este artículo presenta una técnica mejorada que genera un código Freeman a partir de caracteres árabes manuscritos. Este código proporciona la longitud de código más corta sin perder información sobre los caracteres, teniendo en cuenta todas las variantes de caracteres árabes manuscritos. Probamos este código utilizando un conjunto de caracteres árabes en varios formatos para identificar caracteres árabes con el fin de aprovechar el código generado por nuestro método mejorado. También realizamos una comparación entre nuestro código Freeman y los códigos generados en otras investigaciones relacionadas. En vista de ello, el código que obtuvimos representa correctamente la letra árabe en todas sus variantes, incluidas las que los algoritmos de publicaciones anteriores no tuvieron en cuenta. En consecuencia, nuestro novedoso método basado en la codificación Freeman representa un avance significativo en el reconocimiento de caracteres árabes. Además, nuestro método permite identificar y presentar correctamente los caracteres árabes.

Palabras clave: Código de Cadena Freeman; Caracteres Árabes Manuscritos; Código de Normalización.

INTRODUCTION

Writing recognition is crucial for many kinds of parts of society, such as automating translation procedures

and archiving documents from the past. The recognition of handwritten Arabic characters is a challenging task due to the variety and cursive characteristics of handwritten Arabic characters. Currently, there are several methods in place for this reason. Latin character recognition has, nonetheless, received a great deal more study and attention than Arabic character recognition, despite the latter’s importance.^(1,2,3,4) Arabic is a language with about 422 million speakers,⁽⁵⁾ which is important since it highlights the critical need for improvements in Arabic character recognition systems. Also, Arabic is both a Semitic language and the fourth most spoken language in the world.⁽⁶⁾

New approaches to enhancing Arabic character recognition have been introduced as a result of ongoing advancements in technology. Of them, the Freeman code is a particularly interesting method that has been demonstrated to be effective and has advanced the area of Arabic character recognition significantly.

We present in this research an improved Freeman encoding specifically for Arabic character recognition. Our method aims to improve the precision of the current recognition algorithms and address the particular problems caused by the complex structure of Arabic characters.

The article is structured as follows: Following the introduction, we go into the features of Arabic writing in the second section. We also go into the various phases of a handwritten character recognition system, focusing on the extraction of features and related work. Our suggested approach is presented in the third section. The fourth section contains the test results. We arrive at our conclusion at the end.

Recognition of handwritten Arabic

Arabic writing differs among writing systems because of a number of special features.^(7,8) Some of the important features that make Arabic writing distinctive are explained in the following:

- The Arabic alphabet consists of 28 distinctive letters. Every character has a unique shape, depending on where it appears in the word and the style of font being used. Every character might have between two and four distinct forms,⁽⁹⁾ as seen in figure 1.
- Arabic is written from right to left, in contrast to many other languages that are written from left to right.⁽¹⁰⁾
- One of the main features that distinguishes Arabic script from other writing systems is the dot. More than half of all Arabic characters include one or three dots, which can be placed at the top, middle, or bottom of the character. These dots are necessary in order to differentiate between characters⁽¹¹⁾ as seen in figure 2.⁽¹²⁾
- The Arabic alphabet is characterized by a number of letters that have looping shapes, as seen by the character ط.
- Arabic script has a large number of bonds, which are complex letter combinations that improve the written language's aesthetics. These ligatures add to Arabic calligraphy's elaborate and artistic style.
- Arabic is classified as a language that uses a cursive script, which is distinguished by the way that letters are combined to form words.

Name	Isolate	Beginning	Middle	end
Alif	ا	أ	آ	آ
Baa	ب	بـ	بـ	بـ
Taa	ت	تـ	تـ	تـ
Thaa	ث	ثـ	ثـ	ثـ
Jeem	ج	جـ	جـ	جـ
Haa	ح	حـ	حـ	حـ
Khaa	خ	خـ	خـ	خـ
Dall	د	دـ	دـ	دـ
Dhaal	ذ	ذـ	ذـ	ذـ
Raa	ر	رـ	رـ	رـ
Zaay	ز	زـ	زـ	زـ
Seen	س	سـ	سـ	سـ
Sheen	ش	شـ	شـ	شـ
Saad	ص	صـ	صـ	صـ
Daad	ض	ضـ	ضـ	ضـ
TTaa	ط	طـ	طـ	طـ
Dhaa	ظ	ظـ	ظـ	ظـ
Ayn	ع	عـ	عـ	عـ
Ghyan	غ	غـ	غـ	غـ
Faa	ف	فـ	فـ	فـ
Qaaf	ق	قـ	قـ	قـ
Kaaf	ك	كـ	كـ	كـ
Laam	ل	لـ	لـ	لـ
Meem	م	مـ	مـ	مـ
Noon	ن	نـ	نـ	نـ
Haa	هـ	هـ	هـ	هـ
Waw	و	وـ	وـ	وـ
Yaa	ي	يـ	يـ	يـ

Figure 1. Alphabet character and their forms

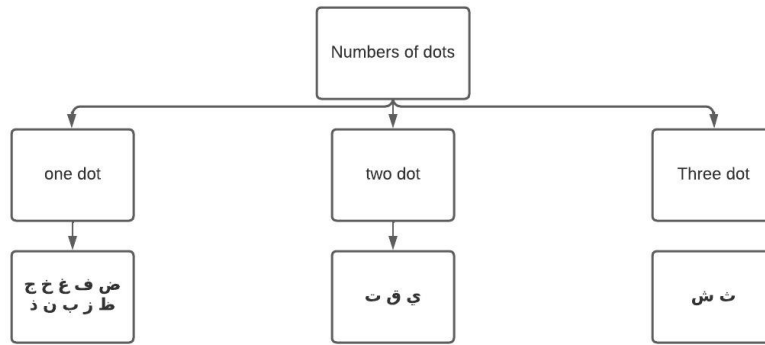


Figure 2. Arabic letters with diacritic points

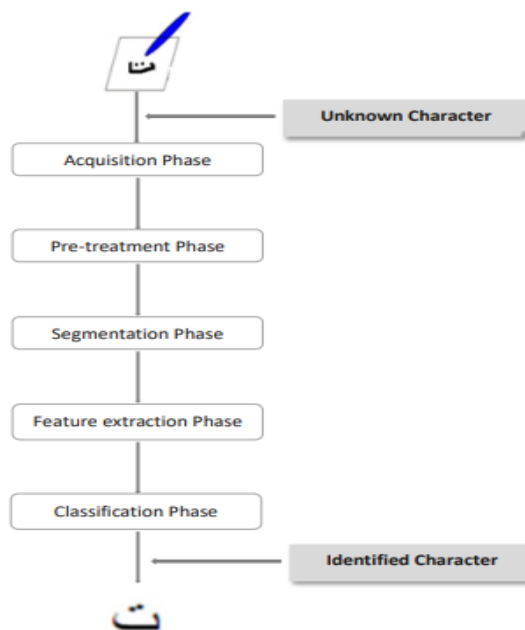


Figure 3. A Standard Recognition of Handwriting System

Handwritten Arabic recognition is the term used to describe the process of recognizing the Arabic letters or words that are present in a digital picture of handwritten writing. A standardized structure, including many essential elements forms the foundation of a handwriting recognition system's conventional structure. This structure usually consists of, as seen in figure 3.

- Acquisition Phase: a crucial stage in the recognition process is the acquisition phase, which involves converting a paper document into a digital picture. This phase consists of several important parts that work together to make the handwriting recognition system successful generally.
- Pre-treatment phase: by reducing noise, improving data quality, and setting the scene for future phases, the pre-treatment phase is critical to the handwriting recognition process. A variety of crucial actions are performed at this phase with the objective of improving the input data specifically, the handwritten Arabic language.
- Segmentation phase: a crucial step in the handwriting recognition process is the segmentation phase, which focuses on dividing every line of handwritten text into separate parts like words or symbols. Several crucial procedures are involved in this phase, all of which help to accurately isolate specific components for further examination.
- Feature extraction phase: the most important step in the handwriting recognition process is the feature extraction phase, in which the forms from the segmentation process are converted into a coded form that captures the most important elements of each segment. In this phase, distinctive characteristics are extracted and used as a foundation for pattern recognition and further analysis.⁽¹³⁾
- Classification phase: a crucial stage in the handwriting recognition process is the classification phase, which involves classifying or categorizing handwritten words or letters according to their visual attributes in a methodical manner. In order to provide the system the ability to analyze and comprehend

the material, this step uses machine learning algorithms or pattern recognition techniques to assign predetermined classes to the retrieved data.

Successfully recognizing characters in optical character recognition (OCR) systems depends on the correct building of feature extraction, a crucial and difficult operation. This process is aided by four primary categories of characteristics: global transformations, statistical characteristics, structural features, and correlation and model superposition.⁽⁶⁾ Moreover, feature extraction techniques are further influenced by the differences between offline and online OCR systems,⁽¹⁴⁾ as seen in figure 4.

In both offline and online OCR, the Freeman code is still a useful technique, particularly for offline systems handling static pictures. It is useful for distilling the essence of characters because of its concise depiction of directional shifts. In contrast, depending on the temporal characteristics of handwriting, online OCR could include extra dynamic elements to improve recognition accuracy. Essentially, the feature extraction method is made for the unique needs of both online and offline OCR systems, ensuring top recognition performance across a range of scenarios.

We will review some research related to the Freeman coding method in the section that follows.

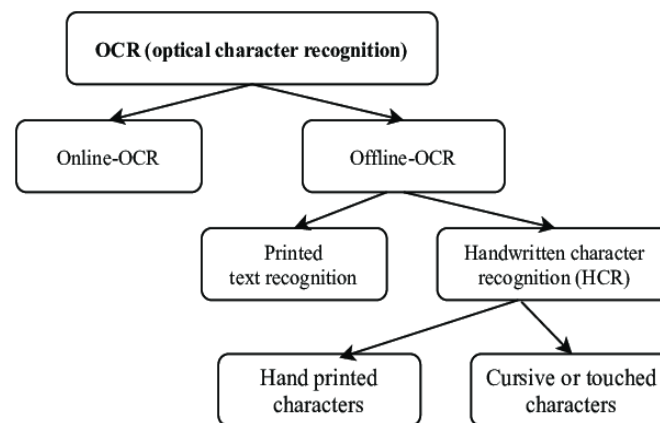


Figure 4. Multiple types of optical character recognition systems⁽¹⁵⁾

It is considered an innovative and extensive extension of previous research that used a similar strategy for Arabic character recognition. One noteworthy work in this field was conducted in 2017 and published in a study.⁽¹⁶⁾ It presented a method for handwritten Arabic character identification using Freeman coding. As discussed in a study,⁽¹⁷⁾ another pertinent work used an Optical Character Recognition (OCR) method to identify isolated Arabic letters by combining template matching with the Freeman chain code. In addition,⁽¹⁸⁾ introduced a technique that was especially designed to identify isolated Farsi/Arabic letters. It included a chain coding algorithm in addition to other characteristics like holes, auxiliary components, and dots.

Furthermore,⁽¹⁹⁾ offers a technique that combines the Freeman code with structural information to determine similarity in written characters. Therefore, we shall outline our technique in the section that follows.

Since the study, other previous studies have achieved the same objective utilizing different methods of characteristic extraction. The Hough transform is suggested for use in the work of Fakir et al.⁽²⁰⁾ in order to recognize written Arabic characters.

Touj et al.⁽²¹⁾ suggested a comparable method that generalizes the Hough transform for the identification of written Arabic letters.

The Hough transform was used to hide Markov models and artificial neural networks by Amor et al.⁽²²⁾

From the analysis of previous work, it is clear that the Freeman method stands out as more efficient than other feature extraction approaches. As a result, we chose to adopt the Freeman code in our methodology, thus recognizing its superior potential for feature recognition in comparison with other existing methods.

METHODS

Pretreatment

In the field of optical character recognition (OCR), the preprocessing process plays a crucial role in preparing documents for effective analysis. One of the fundamental steps in this preprocessing is the conversion of the original image into a binary representation, followed by filtering to optimize image quality. The conversion to a binary image simplifies the complexity of the image by keeping only the binary values of black and white, thus creating a clear basis for subsequent character identification. At the same time, the application of filters aims to reduce imperfections and unwanted noise, thus improving the sharpness of the character contours.

Gaussian blur filter

A Gaussian blur filter is a technique commonly used in image processing to reduce unwanted noise and detail by applying a convolution operation with a gaussian function. This filter is particularly effective for softening or blurring an image while preserving its overall structure. It is often used in image editing and editing software to improve the visual quality of photographs. As seen in figure 5.



Figure 5. Example of Gaussian Blur Filter Application

Canny Filter

The Canny Filter is a contour detection filter widely used in image processing and computer vision. It was developed by John F. Canny in 1986 and is known for its ability to effectively detect contours while minimizing response to noise variations. Canny’s filter goes through several key steps to achieve its goal, including noise reduction, gradient calculation, local non-maxima suppression, and contour detection by thresholding with hysteresis. As seen in figure 6.



Figure 6. Example of Canny Filter Application

Extraction of Feature

The main field of study in the area of automated Arabic language recognition is the exploration of techniques for character extraction. In order to represent Arabic letters in a precise and numerical manner, this method necessitates a careful examination of the form curves and distinctive points that are inherent in the characters. In this work, we presented a novel method based on the addition of a four-element vector.

The p1 element of this vector holds a crucial value, expressing the number of characteristic points present in each character. As for the p2 element, it is used to record the position of these points, cleverly using the value 1 to indicate a point above the character, the value 2 to mean a point below, and the value 0 to indicate the absence of a point. The third element, p3, attempts to identify the number of holes in the character. Thus, the standardized code of the Arabic character is recorded in the fourth element of the vector.

$$P = [P1, P2, P3, \text{normalized code}] \quad (1)$$

As a seen in example:



$$P = [0, 0, 1]$$

Focusing on how important it is to employ this vector is essential, especially in order to clarify the inherent confusion between characters like "م" and "ه", as seen in figure 7. This creative technique elegantly addresses the issue that arises often with Freeman's code, which is restricted to supplying the body code of the main character. This approach not only improves the technical efficacy but also the dependability of Arabic character identification, making a substantial contribution to the advancement of research in this particular sector.

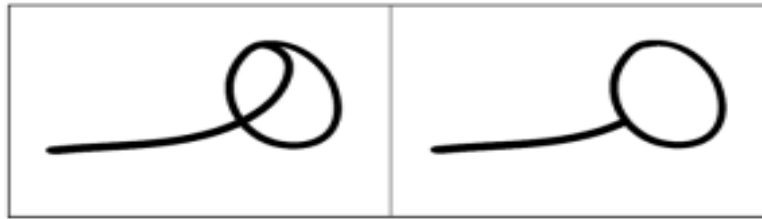


Figure 7. The character "ا" has its original body on the left, and its reconstructed body from the code provided by the previous works algorithm on the right. It is significant that the reconstructed body resembles⁽²³⁾ that of the character "م"

Freeman chain code

We introduce a novel technique to recognize individual handwritten Arabic letters that is based on the Freeman chain code. One method of representing object contours in an image is the Freeman chain code, often known as Freeman coding. Form recognition and computer vision are the two main applications for it.

The basis of Freeman's coding, a popular method in computer vision and image processing, is the idea that a contour may be represented as a carefully organized collection of directions. This encoding technique uses a collection of discrete directions to reflect the pixels that lie around a particular position in a picture. These directions typically cover a range of possible motions within an 8-neighbor grid. The directions on the grid are labeled numerically, ranging from 0 to 7. Each number indicates a different direction.

Figure 8 probably depicts the precise translation of these directions in the context of Freeman's coding, offering a visual aid for comprehending how the ordered series translates to the movement of pixels along the contour. Applications like pattern analysis, edge detection, and form identification benefit from this strategy.

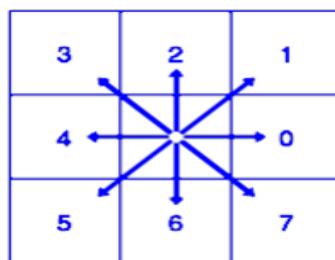


Figure 8. The Freeman coding's 8-connectivity method, which shows each pixel's orientation counterclockwise

Freeman's coding is a crucial professional technique in image analysis that helps to standardize and simplify contour representation, which facilitates the processing and interpretation of visual data by computers. Its effective encoding method makes a variety of computing tasks easier and advances areas such as computer vision and picture processing. Freeman's coding is used by academics and industry professionals to improve the precision and effectiveness of image-based applications, highlighting its importance in the larger field of digital image processing.

The first step in utilizing Freeman encoding to represent a contour is choosing a beginning point on the contour. The next step is to save the instructions from each phase and follow the contour. The Freeman chain, which depicts the contour, is made up of the recorded direction sequence. As seen in figure 9.

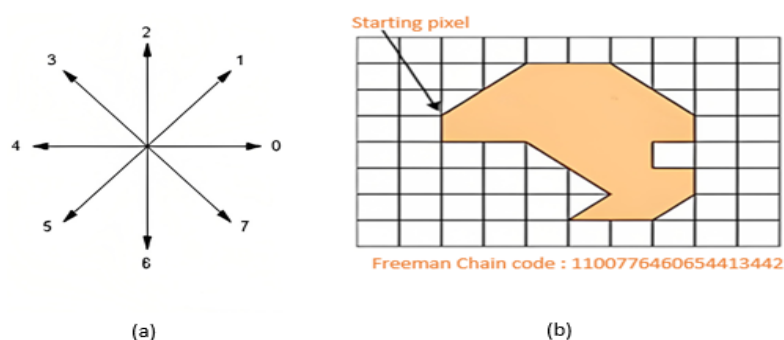


Figure 9. (a) 8-connectivity of the Freeman chain code. (b) An example of an "8-connectivity"

Freeman Code Normalization

Following the extraction of the Arabic character's Freeman code, this code has to be normalized. Our

approach thus depends on limiting the length of the offered Freeman code. Moreover, these are the primary steps we used in our approach:

- Calculating the code length.
- Length Adjustment:
 - Zeros are appended to the end of the code to make it ten characters long if the code length is less than ten.
 - The code is shortened to keep just the top 10 most often occurring directions if the code length exceeds 10.
- Direction frequency calculation.
- Selecting the most frequent directions:
 - The top ten most often offered directions are chosen after the directions are arranged in descending order of frequency.
- Reorganizing the code with specific directions:
 - The modified Freeman code contains only the selected directives.
- Zeros are added to get a fixed length.

In summary, the suggested method provides a consistent and uniform representation of Arabic characters using the Freeman code by defining a set length of 10 characters. This standardization can be achieved by inserting zeros at the end of the code or, if the intended length is exceeded, by choosing the most frequent directions to preserve consistency. By enforcing this fixed-length limitation, the method not only solves potential variances in character representations but also simplifies the following processing phases, resulting in a more robust and efficient recognition system. This rigorous treatment of the Freeman code's length adds to the overall dependability and precision of Arabic character recognition in a variety of settings and applications. As seen in figure 10.

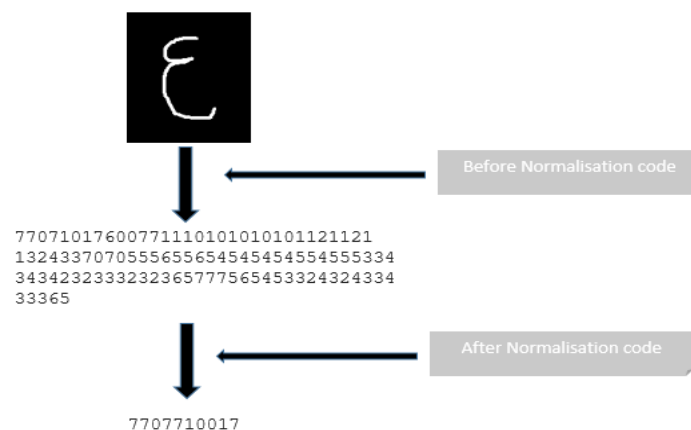


Figure 10. Example of our approach to normalization

RESULTS

To evaluate the effectiveness of our Freeman coding algorithm in the recognition of Arabic letters, experiments were conducted. We conducted an experiment using the AHCD database, which contains 16 800 Arabic letters divided into 28 classes. Each participant wrote each character (from alef to yeh) 10 times. The database is divided into two sets: a learning set of 13 440 characters with 480 images per class, and a test set of 3 360 characters with 120 images per class. This methodology allowed us to test our Freeman coding algorithm exhaustively on a representative database, thus offering a thorough evaluation of its performance in the recognition of Arabic letters.

Two results were attained as a result of the vector utilized in the feature extraction process:









- First result: utilizing the first three vector properties, categorize characters into teen groups. Arabic character categorization is seen in table 1.
- Second result: since the method only provides the Freeman code for the Arabic character's perimeter, we may resolve the issue of confusion between the «ا» and the «آ» characters during the recognition phase by counting the number of holes in each letter. As a result, we discovered one hole for the letter «ا» and two for the letter «آ».

Normalizing the Freeman code is essential, as we previously discussed. Next, the freeman code normalized for a few handwritten Arabic letters is displayed in table 2.

Table 1. Arabic Characters Divided into Teen Groups

	P= (P1, P2, P3)	Character
1	P= (0, 0, 0)	س ط ر ع د ل ك ص ح ا
2	P= (1, 1, 0)	ن ذ ز غ خ
3	P= (1, 2, 0)	ب ج
4	P= (1, 1, 1)	ف ظ ض
5	P= (2, 2, 0)	ي
6	P= (2, 1, 0)	ت
7	P= (2, 1, 1)	ة ق
8	P= (3, 1, 0)	ث ش
9	P= (0, 0, 2)	هـ
10	P= (0, 0, 1)	هـ و

Table 2. Results Achieved by Using Handwritten Arabic Characters

Arabic character	Image of Arabic character	Normalized Freeman code
أ		3407521600
ب		3726405100
ح		3762041500
د		3104527600
ر		3704261000
س		0734652100
ص		3702465100
ط		4057312600

ع		6702531400
ف		4037651200
ك		7362041500
ل		6215374000
م		7306241500
ن		5621430700
ه		5246130700
و		3470652100
ي		4015732600

Table 3. Results of Application of the Recommended Method

Image of Arabic character	Normalized Freeman code
	2615374000
	2613574000
	2651374000

Three instances of the «ل» character written in different formats are shown in table 3, along with their normalized Freeman codes. The results illustrate that, in spite of the variations in writing, our normalization approach produces the same code.

CONCLUSION

Our study concentrates on the exact extraction of characters from Arabic handwriting. As a result of this, we created a novel approach that mixes standard Freeman code with diacritic characteristics. Our study represents a significant step forward in the standardization of Freeman code, thus strengthening the effectiveness of our method. We managed to categorize the letters into ten distinct groups using diacritic characteristics. In addition, we solved the problem of confusion between the characters «ا» and «آ» during the recognition phase. In addition, our standardization method has produced encouraging results. This innovative approach promises to significantly improve the accuracy of extracting characters from Arabic handwriting. This paves the way for practical applications in fields such as optical character recognition and document scanning.

REFERENCES

1. M. AbdElNafea et S. Heshmat, « Novel Databases for Arabic Online Handwriting Recognition System », in 2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE), 2020, p. 263-267.
2. Byerly, T. Kalganova, et I. Dear, « No routing needed between capsules », *Neurocomputing*, vol. 463, p. 545-553, nov. 2021, doi: 10.1016/j.neucom.2021.08.064.
3. V. Jayasundara, S. Jayasekara, H. Jayasekara, J. Rajasegaran, S. Seneviratne, et R. Rodrigo, « TextCaps : Handwritten Character Recognition with Very Small Datasets », in 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), janv. 2019, p. 254-262. doi: 10.1109/WACV.2019.00033.
4. Baldominos, Y. Saez, et P. Isasi, « A Survey of Handwritten Character Recognition with MNIST and EMNIST », *Applied Sciences*, vol. 9, no 15, Art. no 15, janv. 2019, doi: 10.3390/app9153169.
5. N. Boudad, R. Faizi, R. Oulad Haj Thami, et R. Chiheb, « Sentiment analysis in Arabic: A review of the literature », *Ain Shams Engineering Journal*, vol. 9, no 4, p. 2479-2490, déc. 2018, doi: 10.1016/j.asej.2017.04.007.
6. M. Alheraki, R. Al-Matham, et H. Al-Khalifa, « Handwritten Arabic Character Recognition for Children Writing Using Convolutional Neural Network and Stroke Identification », *Human-Centric Intelligent Systems*, vol. 3, nov. 2022, doi: 10.1007/s44230-023-00024-4.
7. N. A. Jusoh et J. M. Zain, « Application of Freeman Chain Codes: An Alternative Recognition Technique for Malaysian Car Plates », 2009.
8. Alaei, U. Pal, et P. Nagabhushan, « Dataset and Ground Truth for Handwritten Text in Four Different Scripts », *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 26, 2012.
9. Alaei, U. Pal, et P. Nagabhushan, « Dataset and Ground Truth for Handwritten Text in Four Different Scripts », *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 26, oct. 2012, doi: 10.1142/S0218001412530011.
10. K. Addakiri et M. Bahaj, « On-line Handwritten Arabic Character Recognition using Artificial Neural Network ».
11. M. S. Khorsheed, « Off-Line Arabic Character Recognition - A Review », *Pattern Anal Appl*, vol. 5, no 1, p. 31-45, 2002.
12. Bataineh, « A Printed PAW Image Database of Arabic Language for Document Analysis and Recognition », *J. ICT Res. Appl.*, vol. 11, no 2, p. 200, août 2017, doi: 10.5614/itbj.ict.res.appl.2017.11.2.6.
13. N. Islam, Z. Islam, et N. Noor, « A Survey on Optical Character Recognition System ». arXiv, 3 octobre 2017. doi: 10.48550/arXiv.1710.05703.

14. S. Djaghbellow, A. Bouziane, A. Attia, et Z. Akhtar, « A Survey on Arabic Handwritten Script Recognition Systems », IJAIML, vol. 11, no 2, p. 1-17, juill. 2021, doi: 10.4018/IJAIML.20210701.0a9.
15. R. Dey et C. Rakesh, « A Novel Sliding Window Approach for Offline Hand-written Character Recognition », déc. 2019, p. 178-183. doi: 10.1109/ICIT48102.2019.00038.
16. H. Althobaiti et Chao Lu, « A survey on Arabic Optical Character Recognition and an isolated handwritten Arabic Character Recognition algorithm using encoded freeman chain code », in 2017 51st Annual Conference on Information Sciences and Systems (CISS), Baltimore, MD, USA: IEEE, 2017, p. 1-6.
17. M. Ali, « Freeman Chain Code Contour Processing for Handwritten Isolated Arabic Characters Recognition », SSRN Electronic Journal, 2012.
18. H. Izakian, S. A. Monadjemi, B. T. Ladani, et K. Zamanifar, « Multi-Font Farsi/Arabic Isolated Character Recognition Using Chain Codes », International Journal of Computer and Information Engineering, vol. 2, no 7, p. 2315-2318, 2008.
19. N. Lamghari, M. E. H. Charaf, et S. Raghay, « Template Matching for Recognition of Handwritten Arabic Characters Using Structural Characteristics and Freeman Code », International Journal of Computer Network and Information Security, vol. Vol. 14 No. 12, 2016.
20. M. Fakir, M. M. Hassani, et C. Sodeyama, « On the Recognition of Arabic Characters Using Hough Transform Technique », Malaysian Journal of Computer Science, vol. 13, no 2, Art. no 2, déc. 2000.
21. Romero-Carazas R. Prompt lawyer: a challenge in the face of the integration of artificial intelligence and law. *Gamification and Augmented Reality 2023*;1:7-7. <https://doi.org/10.56294/gr20237>.
22. Gonzalez-Argote J. A Bibliometric Analysis of the Studies in Modeling and Simulation: Insights from Scopus. *Gamification and Augmented Reality 2023*;1:5-5. <https://doi.org/10.56294/gr20235>.
23. Gonzalez-Argote D, Gonzalez-Argote J, Machuca-Contreras F. Blockchain in the health sector: a systematic literature review of success cases. *Gamification and Augmented Reality 2023*;1:6-6. <https://doi.org/10.56294/gr20236>.
24. S. Touj, N. B. Amara, et H. Amiri, « Generalized Hough Transform for Arabic Printed Optical Character Recognition », vol. 2, no 4, 2005.
25. N. Ben Amor et N. Essoukri Ben Amara, « Multifont Arabic Characters Recognition Using Hough Transform and Neural Networks », in *Advances in Neural Networks - ISNN 2006*, vol. 3972, J. Wang, Z. Yi, J. M. Zurada, B.-L. Lu, et H. Yin, Éd., in *Lecture Notes in Computer Science*, vol. 3972. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, p. 293-298. doi: 10.1007/11760023_43.
26. M. Nasri et M. Kadi, « A modified algorithm for improving the coding of Arabic characters by the Freeman chain code », in 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), Fez, Moroc-co: IEEE, avr. 2019, p. 1-5. doi: 10.1109/WITS.2019.8723747.

FINANCING

The authors did not receive financing for the development of this research.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

AUTHORSHIP CONTRIBUTION

Conceptualization: Mohamed Rida Fethi, Othmane Farhaoui, Imad Zeroual, Ahmad El Allaoui.

Research: Mohamed Rida Fethi, Othmane Farhaoui, Imad Zeroual, Ahmad El Allaoui.

Drafting - original draft: Mohamed Rida Fethi, Othmane Farhaoui, Imad Zeroual, Ahmad El Allaoui.

Writing - proofreading and editing: Mohamed Rida Fethi, Othmane Farhaoui, Imad Zeroual, Ahmad El Allaoui.