PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

MOHAMED BOUDIAF UNIVERSITY - M'SILA

Faculty of Mathematics and Computer Science

Computer Science department

No.:……..

DOMAIN: Mathematics and Computer Science

BRANCH: Computer science

OPTION : ARTIFICIAL INTELLIGENCE

A Dissertation
Submitted in partial fulfillment of the requirements for the degree of Master in Artificial intelligence

By: Chiekhaoui Marouane

Ali aichouche

Titled

Recognition of arabic handwritten letters using deep Learning approach

Under the supervision of

Dr. Rahima Bentrcia

Supported before the jury composed of:

Dr. Rahima Bentrcia	University of Msila	Supervisor
	University of Msila	Reporter
	University of Msila	Examiner

Academic year: 2022 / 2023

Contents

LIST OF FIGURES

LIST OF TABLE

Dedication

Praise be to God, enough is enough, and prayers and peace be upon the beloved Chosen One, his family, and those who are loyal, as for what follows

We thank God Almighty who has enabled us to complete this scientific research and who has inspired us with health, wellness and determination.

We extend our sincere thanks and appreciation to the supervising Professor Dr. (Bentrcia Rahima) for all the guidance and valuable information she provided us that contributed to praising the subject of our study. I also thank her diligence in

following up with us during her vacation period. We also extend our sincere thanks to the members of the esteemed discussion committee, Dr. And let us not forget Providing thanks and gratitude to everyone who contributed to our reaching this blessed place...,

We say thank you very much for all your efforts

INTRODUCTION GENERAL

Artificial intelligence (AI) is a groundbreaking technology that has become a prominent field of study in computer science. It aims to replicate intelligent behavior, such as problem-solving, decision-making, and pattern recognition, in machines. Artificial intelligence has contributed to various fields, including natural language processing (NLP), which seeks to create a direct link between humans and machines through natural language communication, either speech or writing.

Writing recognition is crucial in NLP, as it allows written information to be used through modern electronic means. The study of NLP has made great progress in various languages, including Arabic, which is spoken by more than 467 million people in the Arab world and neighboring regions. Despite its wide use, there is a lack of research on Arabic computation, particularly in the field of recognition, due to the complexity and ambiguity of Arabic script [1]

To address this gap, we wanted to develop a handwritten Arabic character recognition system using Convolutional Neural Networks (CNN), which has proven effective in achieving fast and reliable recognition results.

Problem Statement:

We are driven to conduct our research to solve the following problems:

- Addressing issues of difficulty in reading ancient Arabic texts and handwriting. Due to the limitations of images containing text stored in the computer, the Arabic character recognition system can offer a suitable solution.
- In addition, manual data entry using the keyboard is a time-consuming process that can lead to errors.
- Lack of research in this field

So, developing a recognition-based segmentation system for Arabic handwritten words may solve the above problems. However, there are many challenges related to Arabic text include:

- Arabic characters: The Arabic language consists of 28 consonants (or 29 if we include the hamza).
- Multiple character shapes: Each Arabic character has two or four shapes depending on its position within a word (isolated, beginning, middle, and end), resulting in a total of 108 different shapes for the 28 letters. The first column of Table 1 represents the letter's order in the alphabet, the second column represents the letter itself, the third column represents the letter in its isolated form, the fourth column represents the letter at the beginning of the word, the fifth column represents the letter in the middle of the word, and the last column represents the letter at the end of the word.

No	Letter Name [a]	Isolated Form	Initial Form	Medial Form	Final Form	No	Letter Name	Isolated Form	Initial Form	Medial Form	Final Form
1	Alef[b, c]	ى ا	-	-	ى ا	16	Tah	ط	ط	ط	ط
2	Beh	ب	بـ	ـبـ	ـب	17	Zah	ظ	ظ	ظ	ظ
3	Teh[d]	ة ت	تـ	ـتـ	ـة ـت	18	Ain	ع	عـ	ـعـ	ـع
4	Theh	ث	ثـ	ـثـ	ـث	19	Ghain	غ	غـ	ـغـ	ـغ
5	Jeem	ج	جـ	ـجـ	ـج	20	Feh	ف	فـ	ـفـ	ـف
6	Hah	ح	حـ	ـحـ	ـح	21	Qaf	ق	قـ	ـقـ	ـق
7	Khah	خ	خـ	ـخـ	ـخ	22	Kaf	ك	كـ	ـكـ	ـك
8	Dal[b]	د	-	-	ـد	23	Lam	ل	لـ	ـلـ	ـل
9	Thal[b]	ذ	-	-	ـذ	24	Meem	م	مـ	ـمـ	ـم
10	Reh[b]	ر	-	-	ـر	25	Noon	ن	نـ	ـنـ	ـن
11	Zain[b]	ز	-	-	ـز	26	Heh	ه	هـ	ـهـ	ـه
12	Seen	س	سـ	ـسـ	ـس	27	Waw[b]	و	-	-	ـو
13	Sheen	ش	شـ	ـشـ	ـش	28	Yeh	ي	يـ	ـيـ	ـي
14	Sad	ص	صـ	ـصـ	ـص	29	Hamza[e]	ء	ئـ	ـئـ	ـأ
15	Dad	ض	ضـ	ـضـ	ـض						

Table 1 The Different Shapes of Arabic Letters *[2]*

Figure 1 shows the four different forms of the Arabic letter waw (waw) based on its position in the word.

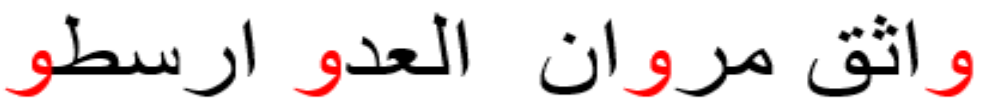

Figure1 The letter (و) in its four positions: the beginning, middle, end of the word, and Isolated.

- The Arabic language is written from right to left

Figure 2 Arabic writing direction

- Dots are of utmost importance in the Arabic language, as some letters are differentiated between them only by dots because they have the same shape, such as ، (غ ، ع) ، (ش ، س) (ث ، ت ، ب) ، (خ ، ح ، ج). Determining the difference according to them based on the position of the dots and their number, because the dots in the Arabic language do not exceed three dots, and we know that half of the Arabic letters are without dots, and the second half is dotted.
- **Connectivity:** Arabic letters are divided into two parts: The Arabic word has separate letters like the Latin letters, and others, on the contrary, are connected, as shown in the Figure 3 and Figure 4.

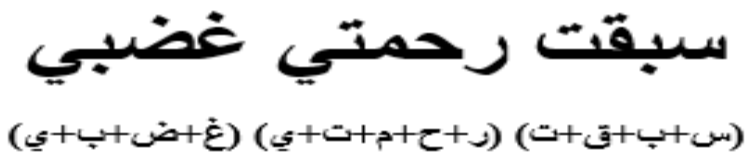

Figure 3 Characters connectivity

Figure 4 Characters Separate

- **Ligatures:** In most Arabic fonts, ligatures occur when two characters overlap within certain syllables of words, as illustrated in Figure 5.

Figure 5 Examples of Arabic Ligatures

- **Diacritics:** Diacritical marks contribute to clarifying the pronunciation of the word as well as its meaning, as clarified in Figure 6.

Figure 6 Arabic Text with Diacritics

- **Overlaps:**

In some Arabic scripts, certain letters in a word may overlap each other, as in the Diwani script that appears in Figure 7.

Figure 7 Some overlapping letters in Arabic fonts.

- **Sub word(s):** As some Arabic words consist of a group of sub-words, Figure 8 shows this.

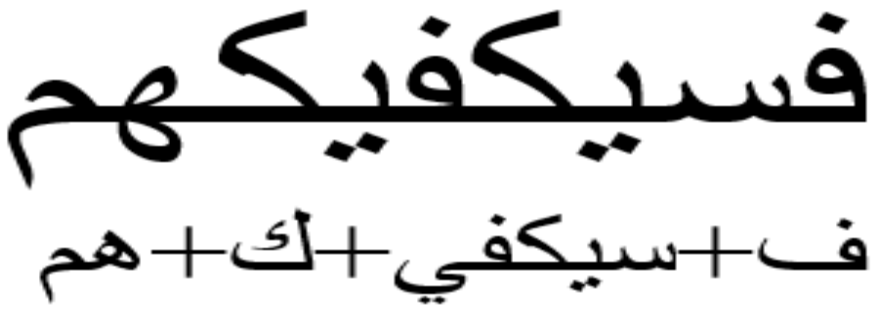

Figure8 Example of Arabic word with four sub-words.

- The Arabic text contains punctuation marks and symbols in its composition, such as (,,.??!)

Objectives of the Research

Since there are 26 Arabic-speaking countries, whether it is an official or unofficial language, and because the number of Arabic speakers has exceeded 400 million, so the Arab world needs researchers and specialists in this field to develop techniques and applications for dividing

The word written in Arabic handwriting, then recognizing the letters and making sure that the division is correct

Research Importance

- Preserving Historical Documents: Handwritten documents, such as archives and historical manuscripts, are often fragile and prone to damage or loss. By digitizing these documents through word segmentation and letter recognition, we can preserve them for future generations and make them accessible to a wider audience.

- Improve accessibility: Segmentation and handwriting recognition technology can help people with disabilities who find it difficult to use traditional input methods, such as keyboards. With a pen or touch screen device, individuals can write by hand and convert their handwriting into digital text

- Multilingual support: Arabic handwriting recognition after word division is just one example of the broader field of recognition for multiple languages. By improving the accuracy of these systems, we can support multilingualism and enable communication across different languages and cultures

Thesis Organization

- In the general introduction, we provide an overview of intelligence and its contribution to the development of linguistic processing, in addition to explaining the status of the Arabic language and its importance among the peoples of the world. The research objectives and importance are also discussed.
- Chapter 1: we describe the dataset used to conduct this research.
- Chapter 2: the main previous works in the field of Arabic text segmentation are highlighted.
- Chapter 3: in this chapter, we propose the recognition system model with word segmentation algorithm.
- Chapter 4: the experimental results are discussed in this chapter
- Conclusion and future work are presented in Chapter 5.

CHAPTER 1: DATASET

1. Introduction

A dataset is a collection of data points related to a specific topic. These data contain a set of information, and may be in the form of images, numbers, audio recordings, or video clips. They can be stored in different formats such as CSV or SQL. Usually, the data are linked to a specific purpose and to the same topic [2]

A dataset is an essential requirement for building the foundation of any artificial intelligence application or project. Therefore, data must be collected in a dataset that contains the appropriate quantity and quality for analysis, so that we can extract hidden information that allows us to make suitable decisions [3]. However, working with data is a complex matter because it requires proper processing of our data to be able to use and divide it.

One of the most important aspects of machine learning is training a model, checking its accuracy, and testing its performance. Therefore, in most cases, we need to split our data into three set in order to make an objective assessment [4]

The training set, also called training data or training set, is the part of our original data that has the largest proportion. It is vital in any machine learning model as it helps you perform the required task and makes accurate predictions [5]Simply put, the training data forms a machine learning model, so you can analyze the data over and over again to understand its intricacies and properties, thereby improving performance.

After the training phase, we have development data, also known as validation data. This set is specifically used for regular assessment during the training phase. The model does not learn from this data, even if it is presented to it from time to time. This validation data helps protect against overfitting and assesses the generalizability of the model. Although validation data is separate from training data, data scientists can keep some of the training data for

validation purposes. Note that the validation data is not involved in the training process as shown in the diagram [5]

Now that we're complete using the validation set, we can proceed to use the test data to approximate the model's performance to actual performance. The testing phase is the final step in evaluating the performance of our model [6]To avoid biased predictions and unreliable results, this set should never be tested before choosing a model. It should be tested as a final shape after training and validation to determine the best model. The three types of sets are presented in Figure 9

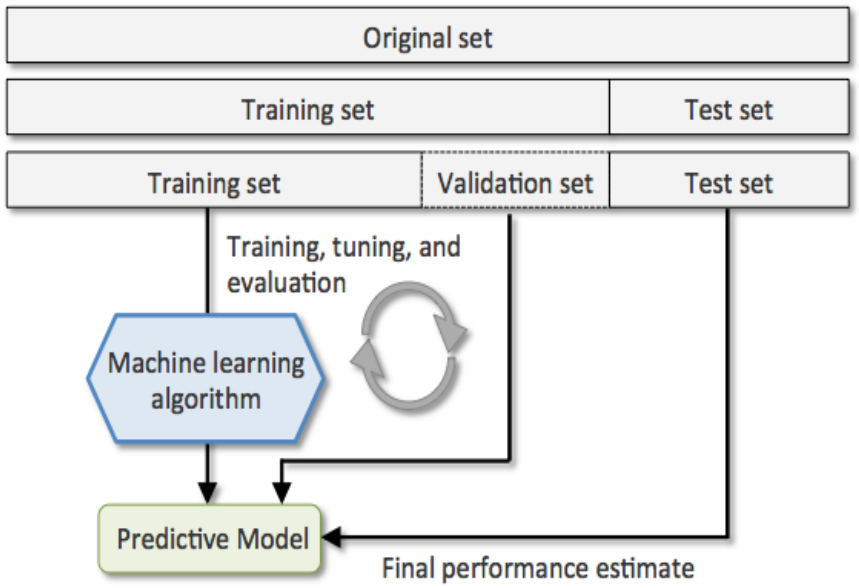

Figure 9 Splitting of a dataset into training, testing, and validation datasets *[7]*

2. AHCD (Arabic Handwritten Characters Dataset):

Datasets are an essential component of any computer vision system. Comparative datasets empower researchers to assess various active and authentic machine learning methods and techniques. In this section, we introduce the AHCD (Arabic Handwritten Characters Dataset), a comprehensive collection of handwritten Arabic characters. These resources are freely available to the public and were obtained from 60 individuals aged between 19 and 40 years, resulting in a meticulously curated dataset of 16,800 characters [8].

3. Data collection

The dataset includes 16,800 handwritten letters crafted by 60 participants aged 19 to 40, with 90% being right-handed. Each participant wrote ten iterations of Arabic alphabet letters ("alif" to "yeh"). Scanned at 300 dpi for detail, Matlab 2016a was used for automatic segmentation.

It's divided into a training set (13,440 characters, 480 images per class) and a test set (3,360 characters, 120 images per class). Writers don't overlap between sets to avoid bias, and random inclusion prevents institutional concentration, ensuring impartial evaluation by various models and algorithms.

4. Data Storage

After completing the data collection, the characters were stored in the following format as shown in Table 2. The first column represents the alphabetical order of the characters. The second column represents the character itself, and the third column represents the name of the folder in which the corresponding character's images were collected. The folder names are designated with Latin letters for pronunciation of the respective character. The fourth column represents the number of images for each character. The last column represents the percentage of images for each category relative to the total number in the dataset.

Table 2 data set storage

num	character	N fils	Name	percenta	num2	characte	N fils2	Name2	percentag
1	أ	480	alif	3.57%	15	ض	480	dad	3.57%
2	ب	480	ba	3.57%	16	ط	480	taa	3.57%
3	ت	480	ta	3.57%	17	ظ	480	daad	3.57%
4	ث	480	tha	3.57%	18	ع	480	ain	3.57%
5	ج	480	gim	3.57%	19	غ	480	gain	3.57%
6	ح	480	haa	3.57%	20	ف	480	faa	3.57%
7	خ	480	khaa	3.57%	21	ق	480	qaf	3.57%
8	د	480	dal	3.57%	22	ك	480	kaf	3.57%
9	ذ	480	thal	3.57%	23	ل	480	lam	3.57%
10	ر	480	raa	3.57%	24	م	480	miim	3.57%
11	ز	480	zay	3.57%	25	ن	480	noon	3.57%
12	س	480	siin	3.57%	26	ه	480	hah	3.57%
13	ش	480	shiin	3.57%	27	و	480	waw	3.57%
14	ص	480	sad	3.57%	28	ي	480	yaa	3.57%

A sample of the AHCD dataset

Figure 3 shown below provides a sample of some random images of the AHCD dataset with different letter [8]

Figure 10 Random samples of the data set

5. Conclusion

Most scientific literature utilizes multiple datasets for various applications. However, the absence of comprehensive reference datasets for printed Arabic characters, covering all Arabic character forms and including overlapping characters, makes it challenging to consistently compare different methods and assess their accuracy. The proposed dataset aims to enrich Arabic language datasets and enable researchers to build more accurate models for character recognition. This dataset comprises 16,800 samples and has the potential for further expansion to enhance its ability to recognize Arabic characters in their various forms.

CHAPTER 2: LITERATURE REVIEW

Introduction:

Character recognition is the technological process that a computer uses to turn printed or handwritten text on pictures into text files. To complete this activity, the computer needs character recognition software. By doing this, the text that is already present in the image may be retrieved and saved in a file that can be utilized in a word processor or kept in a database by a computer system. There are several character recognition engines in use today, some of which are free and some of which cost money, and they all employ different methods and a lot of data to understand the content.

Previous Studies:

Many research and development efforts and investments have been made to address the problem of handwriting recognition in the Arabic language, with the aim of developing advanced technologies that enable more accurate and reliable recognition. In the following lines are some examples of previous work in the field of handwriting recognition.

- Altwaijry and Al-Turaiki in [9]developed an automatic handwriting recognition model that was trained using the hijja dataset, as well as the Arabic Handwritten Characters (AHCD) dataset. The model performance results are as follows: It achieved an accuracy of 97% on the AHCD dataset and 88% on the Hijja dataset, respectively.

- Another research was carried out by El-Sawy et al., [10]which is based on Arabic character recognition using Convolutional Neural Networks (CNN). CNN has better performance in both images and big data. In the experimental section, the results were promising, as the classification accuracy rate reached 94.9% in the image test.

- Most handwriting recognition systems have focused primarily on adult handwriting, with limited research on children's handwriting, which is challenging due to its low quality. In addition, many of these systems designed for adults have not been adapted or tested to recognize pediatric data. In a research worked on by Alwagdani and Jaha, [11], a new convolutional neural network (CNN) model was developed to identify isolated handwritten Arabic letters in children, and includes various datasets from Hijja (child data) and AHCD (adult data). The results underscore the importance of the training approach, as including adult data was shown to boost accuracy to about 93% in recognizing children's handwriting. Moreover, the combination of the proposed complementary features and deep features improves children's handwriting recognition performance by approximately 94%.

- Another study was conducted by Balaha et al., [12]presented 14 different CNN architectures through a series of trial and error experiments. These architectures were trained and evaluated using the HMBD database, which comprises 54,115 handwritten Arabic characters. The initial CNN models achieved a maximum test accuracy of 91.96%. To enhance performance, a novel approach called "HMB-AHCR-DLGA" was introduced, combining transfer learning (TF) and a genetic algorithm (GA) to optimize training parameters and hyperparameters during the recognition phase. This approach incorporated pre-trained CNN models like VGG16, VGG19, and MobileNetV2. After conducting five optimization trials, the best combinations were identified, resulting in the highest reported test accuracy of 92.88.

- In [13], Modhesh and Al-Mudhaffair introduced the VGG Alphanumeric Network for the recognition of handwritten Arabic alphanumeric characters, inspired by the remarkable success of the very deep VGGNet architecture. The proposed model shows improvements in speed and reliability, which ultimately leads to improved classification performance. In addition, it contributes to reducing the overall complexity of the VGGNet architecture.

  To evaluate the effectiveness of the approach, evaluations were conducted on two widely recognized standard databases. A verification accuracy of 99.66% was achieved on the ADBase database and 97.32% on the HACDB database.

Table 3 Previous work on letter recognition

References	Year	Model	Dataset	Type	Accuracy
[9]	2020	CNN	AHCD Hijja	Chars	97% 88%
[10]	2017	CNN	AHCD	Chars	94.9%
[11]	2023	CNN	AHCD Hijja	Chars	94%.
[12]	2021	CNN	HMBD	Chars	92.88%.
[13]	2017	VGGNet	HACDB	Chars	97.32%

Chapter 3: The proposed segmentation and recognition system

1. Introduction:

This chapter reviews the working environment and steps followed to implement a system for segmenting Arabic handwritten words using a deep approach to machine learning to improve the overall performance of this process.

This goal was achieved by implementing a complex convolutional neural network and advanced techniques in the field of deep neural networks and machine learning. The deep machine learning approach involves analyzing and using a training data set containing multiple examples of Arabic handwriting.

In addition, the performance of the developed system has been evaluated accurately and reliably. This aims to provide objective results that reflect the effectiveness of the approach used and the extent to which it achieves the desired goals in dividing Arabic words by hand.

This study represents a serious attempt to provide an advanced technical solution to the problem of dividing handwritten Arabic words, and represents an important progress in the field of machine learning and Arabic language processing.

2. Working environment:

2.1. Programming Language and Tools:

Python:

This project was implemented using the Python programming language due to its powerful capabilities in deep learning and natural language processing [14]Python served as our primary language for coding and interacting with the libraries and tools used in the project.

2.2. Libraries and Frameworks:

TensorFlow:

TensorFlow is an open-source framework used for building and training deep learning models [15]It facilitates the creation of complex neural networks and the execution of machine learning tasks in general.

NumPy:

NumPy is a fundamental library for mathematical operations and numerical computations in Python. It aids in performing mathematical operations efficiently on data and numbers.

Pandas:

Pandas is employed for data analysis and processing. It can be used for reading and writing data from various sources and converting data into suitable formats for training.

OpenCV:

OpenCV is a specialized library for image and video processing. It is used for tasks such as reading, editing, analyzing images, and extracting information from them [15]

Matplotlib:

Matplotlib is used for data visualization and image display [16]It can be utilized to showcase results and create graphical representations.

Tkinter:

Tkinter is a library used for creating graphical user interfaces (GUIs) in Python. It simplifies the design and programming of interactive user interfaces [17]

Keras:

Keras is a high-level interface for building and training deep learning models. In your case, it appears that you are using it to load pre-trained models.

2.3. Hardware and Performance Requirements:

CPU: Intel Core i7 (4th generation)

RAM: 8GB

Storage: 256GB SSD (Solid State Drive)

GPU: AMD Radeon R7

2.4. Data:

A database of 16,800 handwritten Arabic characters, with a training set consisting of 13,440 characters spread over 480 images per class, and a test set containing 3,360 characters spread over 120 images per class [18]

3. Data preprocessing:

Image preprocessing is a crucial series of operations performed on an image file with the objective of enhancing the quality of the images and eliminating redundant information to improve the accuracy of recognition processes. In any image processing application, it is essential to go through the preprocessing step, which involves various procedures like shifting, binarization, resizing, smoothing, and more. The primary aim of the preprocessing phase is to reduce noise levels without altering the essential information within the image.

3.1. converting data to numpy array:

The data from the dataset is converted into a NumPy array. Most machine learning libraries work best with NumPy arrays. The data will be represented as a 3D array where we have the width and height of the images (32 x 32) as a 2D array.

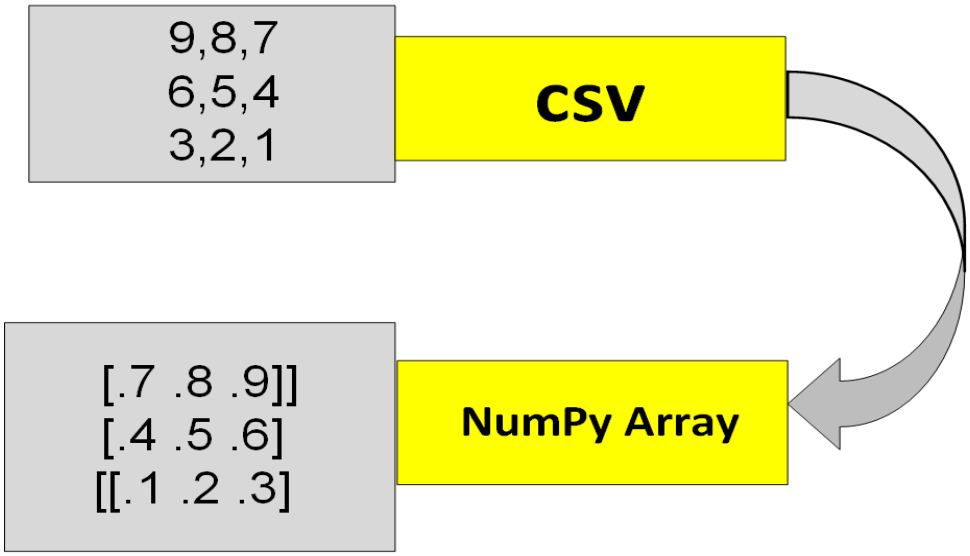

Figure 11 Convert a CSV to NumPy Array

3.2. Image Rotation:

At this stage, the images are processed to perform a specific transformation: rotation.

Each image in the dataset is rotated 90 degrees clockwise using the OpenCV (cv2) library for this task. This means that the images are oriented clockwise.

After rotation, the images are further processed by flipping them horizontally. Rotation and flip are applied to each image in the dataset.

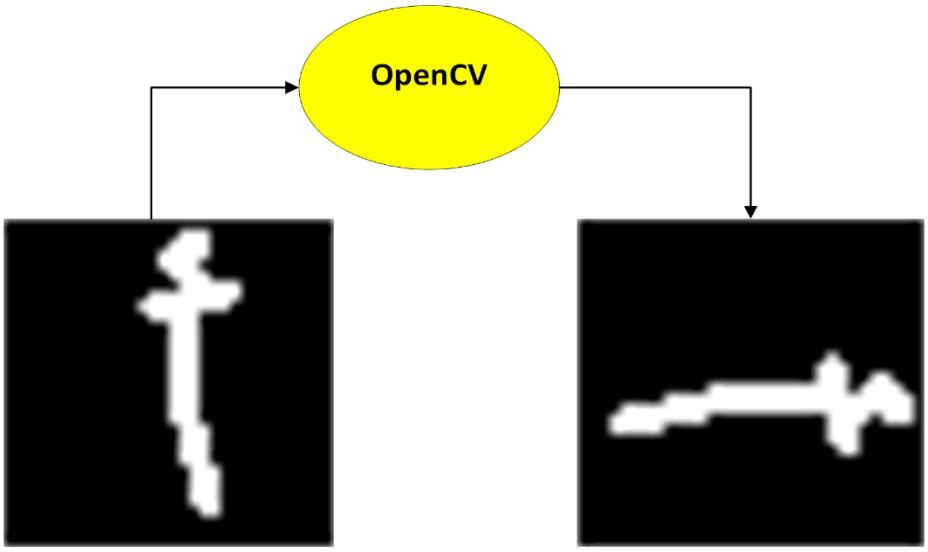

Figure 12 Image Rotation:

3.3. Converting Data to Model-Compatible Format:

After rotation and flipping operations, the shape of the data must be modified to be compatible with the input requirements of the machine learning model or neural network.

An additional dimension of size 1 is added to represent the grayscale channel. This is because the images are now grayscale, and typically in deep learning, a 3D tensor is expected, where the third dimension represents the color channels. In this case, there is only one channel (grayscale), so the format becomes (32x32x1).

3.4. Scaling Data Values:

The values in the data are scaled to a certain range to ensure that they are in an appropriate format for machine learning.

In this case, the values are scaled to be between 0 and 1. This is achieved by dividing each pixel value by 255.

This measurement is important because many machine learning algorithms work best when the input data falls within a certain range.

3.5. Returning Data as a NumPy Array:

Finally, the function returns the preprocessed data as a NumPy array.

The pre-processed data is now in a format ready to be used in training the machine learning model. In short, these stages collectively prepare the input image data for machine learning by converting it to a suitable format, applying the necessary transformations, and scaling it to ensure optimal performance of the model.

4. Proposed recognition model:

The goal of the proposed model is to build a robust and efficient model that can classify handwritten Arabic letters into specific categories (28 categories) with ruffles. This model is distinguished by its focus on using multiple layers of artificial neural networks to extract features from images and improve classification performance. The number of layers in the proposed model is 11 layers. The first layer uses Conv2D type with 32 filters and kernel size (3, 3), where the kernel is activated using the ReLU function. The input size is saved with the approved padding “same”, and this is the same as the rest of the layers in the activation and saving sizes. The input size is (32, 32, 1). MaxPooling2D window size (2, 2) is used to reduce the image size. BatchNormalization is added after each Conv2D layer to ensure the stability of the training process. Then we get to the second and third layers, each of which contains a Conv2D layer, the second after the first layer, and the third after MaxPooling. The second layer contains 64 filters and the third layer contains 128 filters. 0.2 sealant is applied in the third layer. Next comes the last layer which contains Flatten to convert the 3D matrix into a flat matrix. We now move on to the full layer containing 32 units, with the final layer containing 28 units, and the softmax activation function is used to generate classification probabilities. Figure 13 and Table 4 illustrate the model architecture.

Table 4 Architecture of the CNN Model

Layer (type)	Output Shape	Param
conv2d	32, 32, 32	320
max_pooling2d	16, 16, 32	0
batch_normalization	16, 16, 32	128
conv2d_1	16, 16, 64	18,496
max_pooling2d_1	8, 8, 64	0
dropout	8, 8, 64	0
batch_normalization_1	8, 8, 64	256
conv2d_2	8, 8, 128	73,856
max_pooling2d_2	4, 4, 128	0
dropout_1	4, 4, 128	0
batch_normalization_2	4, 4, 128	512
flatten	2048	0
dense	32	65,568
batch_normalization_3	32	128
dropout_2	32	0
dense_1	28	924
Total params:160188 Trainable params:159676 Non-trainable params: 512		

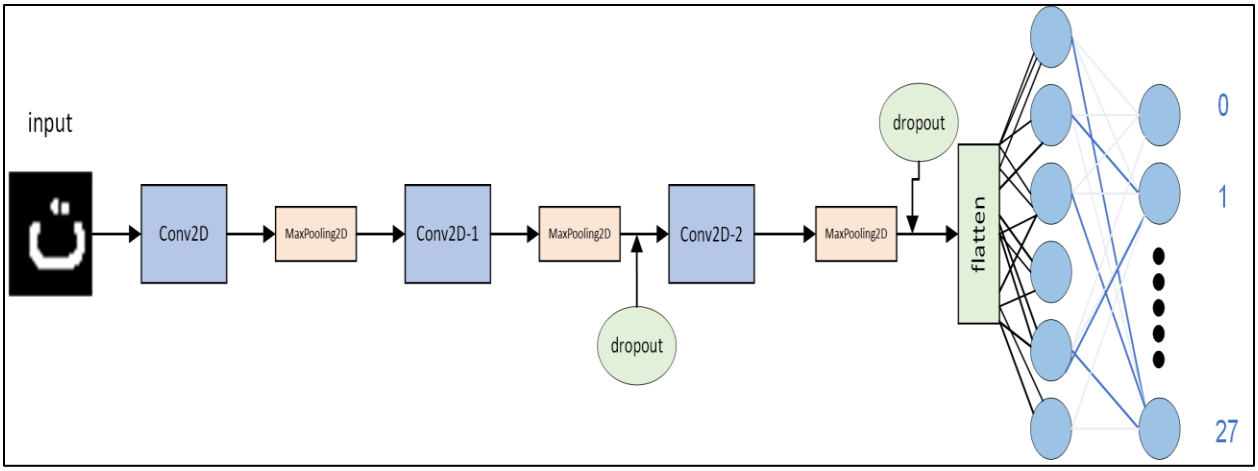

Figure 13 Layer Structure of the CNN

4.1 CNN Model Training

The neural network was trained using a dataset consisting of 16,800 images. The dataset was split into an 80% training set and a 20% validation set. During training, a batch size of 147 was selected, which means that the model's weights were updated after processing every 147 samples. This training process was iterated 600 times to allow the model to learn from the data thoroughly.

The training process yielded impressive results. The accuracy of the training set reached an impressive 99.05%, demonstrating the model's ability to correctly classify images in the training data. Similarly, the accuracy of the validation set was 93.77%, indicating the model's ability to generalize its learning to unseen data.

As the model continued to learn from more examples, both the training and validation losses decreased significantly. The training loss dropped to 6.15%, reflecting the model's improved ability to minimize errors on the training data. Likewise, the validation loss decreased

to 9.3%, indicating that the model was effective at generalizing its learned knowledge to new, previously unseen data.

For a visual representation of the training and validation results, please refer to Figure 14 .

Figure 14 Accuracy and Loss Results of the CNN Model

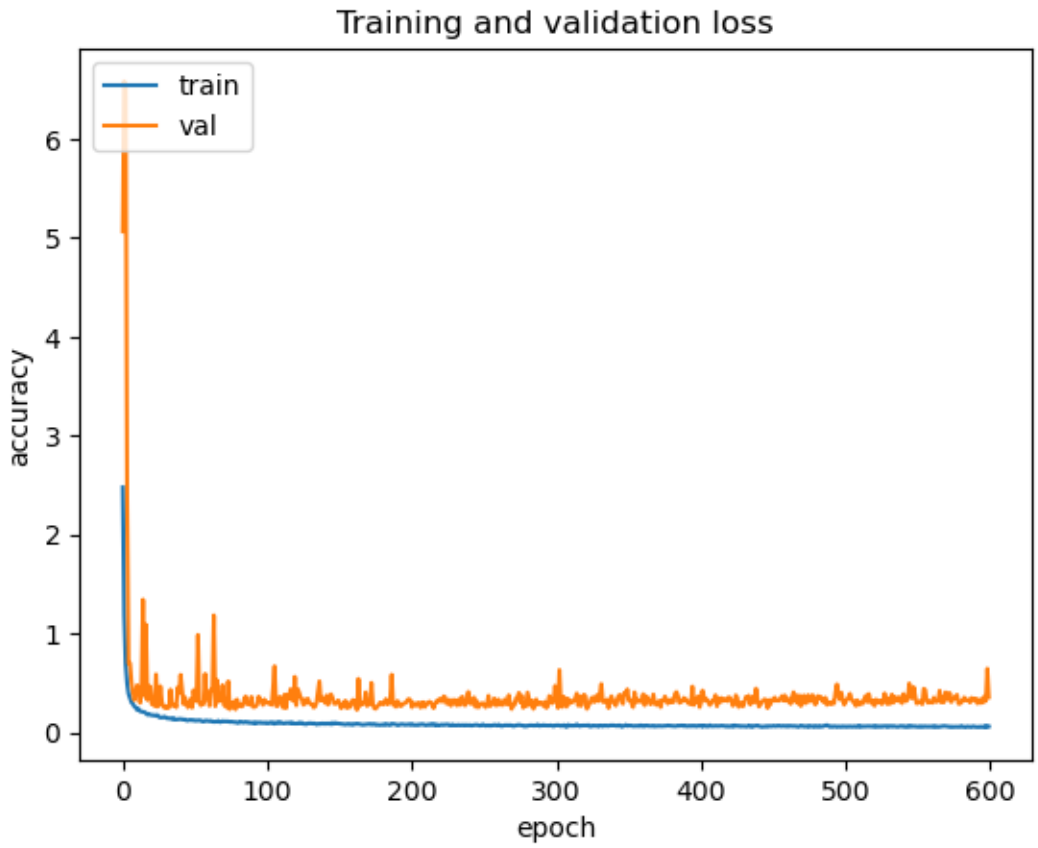

Figure 15 Accuracy and Loss Results of the CNN Model

5 Handwritten Arabic word segmentation:

As we mentioned previously, our research includes segmentation of the Arabic word and then recognition of its letters by means of deep learning. The part related to recognition has been explained and now the part related to division, where in the first step the image of the word is read and converted into a simple gray-scale model image using the open cv library after reading the image. The vertical distribution (vertical graph) is calculated using numpy. The goal here is to know the amount of black pixels (which represent letters) in each column of the image. This helps in determining its location in the image. In this next step, a threshold is defined to differentiate the two characters. The default value of the threshold is set to (threshold = 1) and this value can be modified by the size and properties of the image. The goal is to determine the value below which the segmentation takes place without disturbing the letter. Then comes the segmentation step and discards all columns that do not meet the threshold. Finally, the divided

characters are saved to be recognized by our model that was built for this purpose. Figure 16 describes these steps in the word (أول)

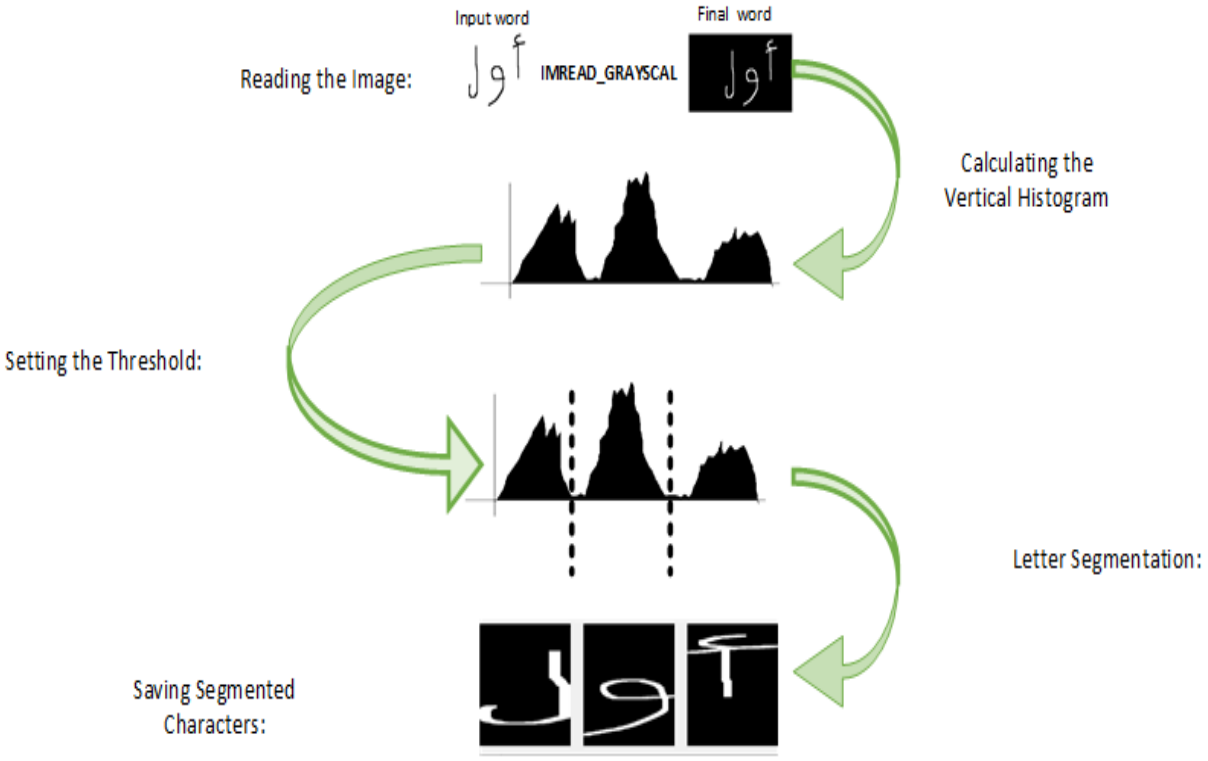

Figure 16 Handwritten Arabic word segmentation

System Interface:

Figures 7, 8, and 9 demonstrate the design of the system interfaces.

The first interface:

The first interface contains a space to segment the handwritten Arabic word into letters, while displaying the segmented images in the same interface.

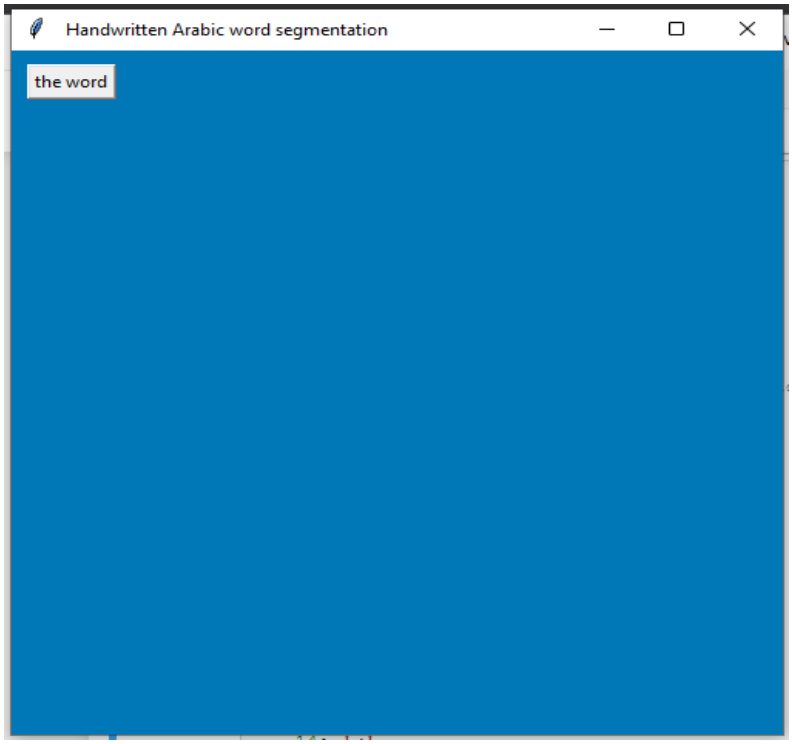

Figure 17 The first interface

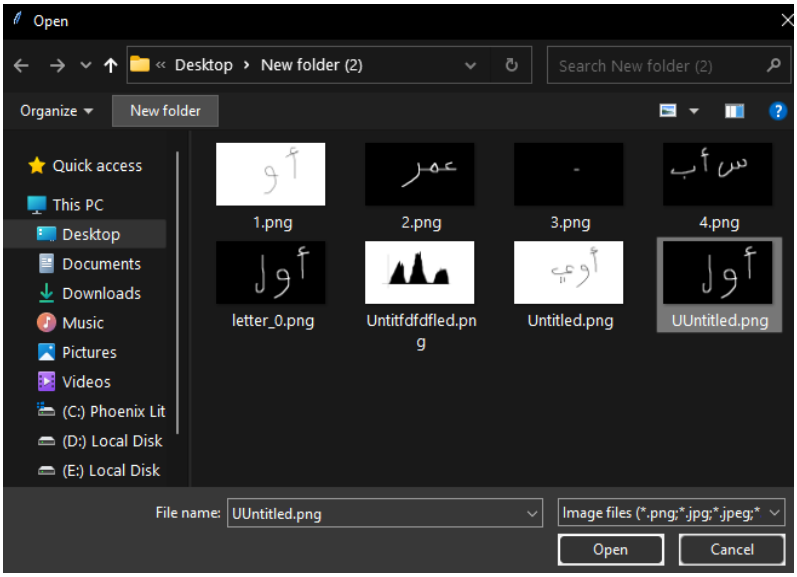

Figure 18 Upload the image

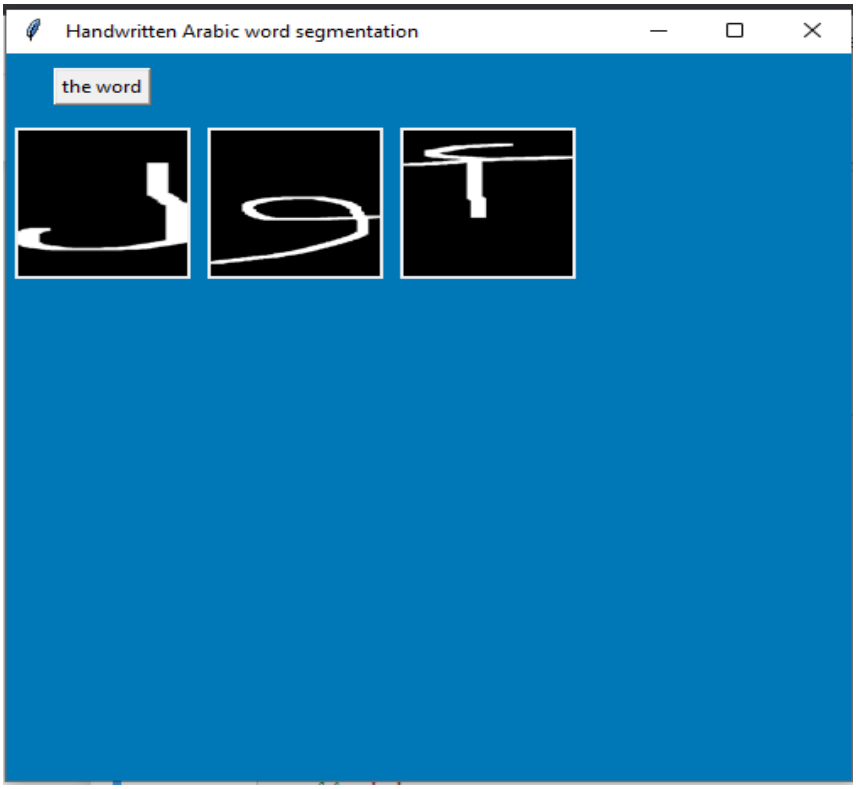

Figure 19 The result

The second interface:

The second interface contains the area where handwritten characters are recognized and the display and prediction are displayed as shown in Figure 10, 11, and 12.

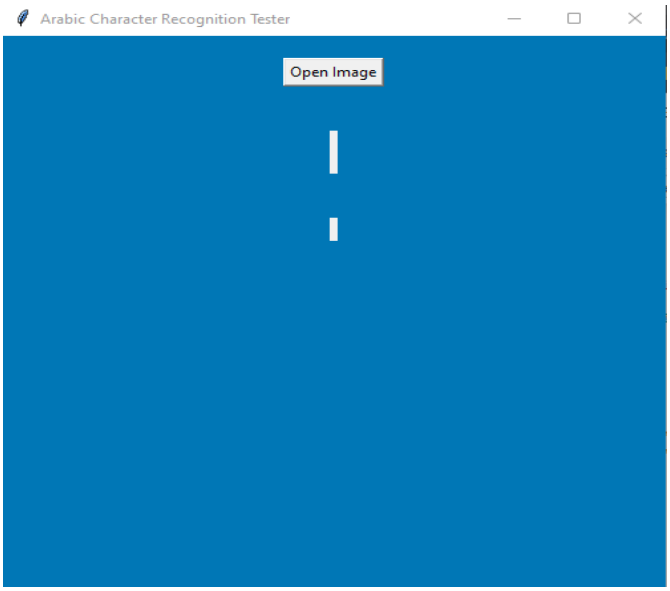

Figure 20 second interface

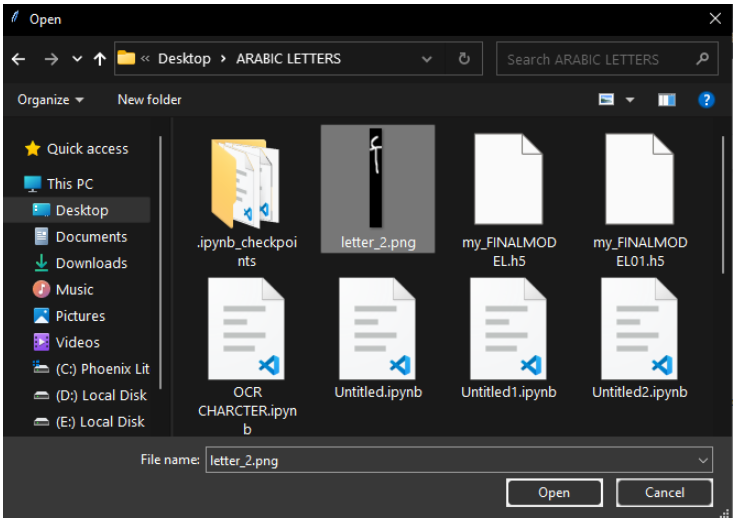

Figure 21 Upload the image

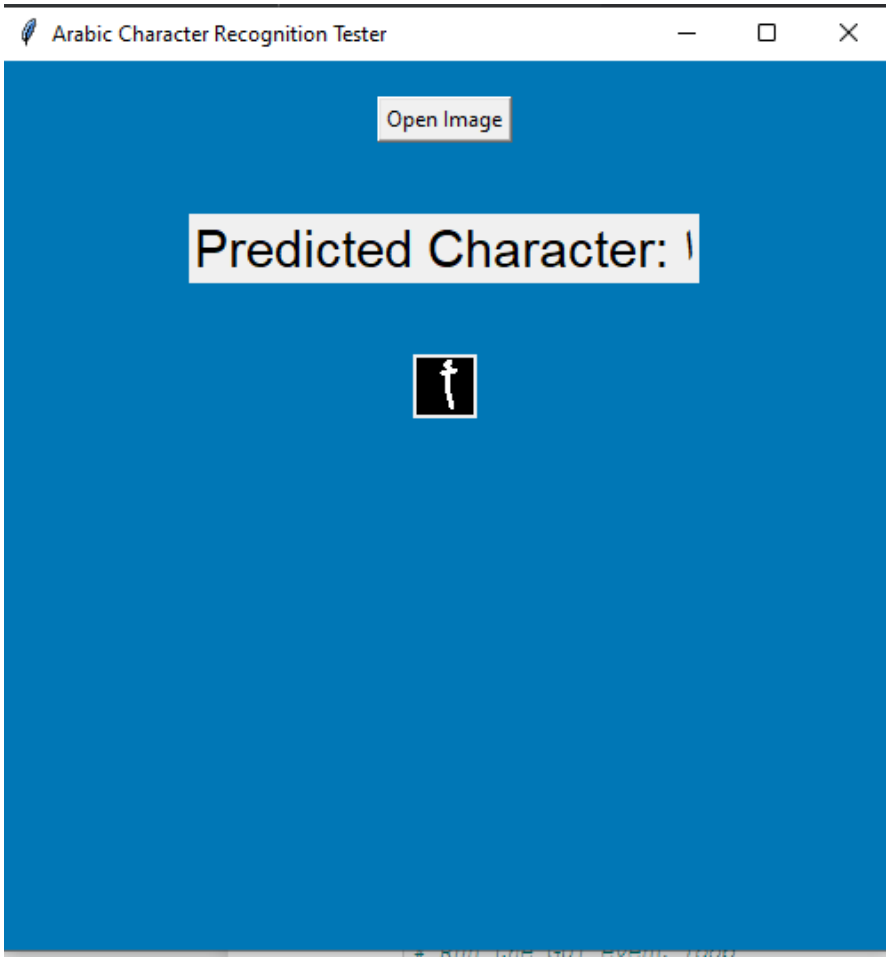

Figure 22 The result

Conclusion:

In this chapter, we have provided an extensive explanation of our deep learning-based approach to segment handwritten Arabic words. Our system consists of four main stages: preprocessing, dividing the word into letters, feature extraction, and classification.

Basically, we used a convolutional neural network (CNN) to extract and classify features. This choice has proven effective in enhancing the recognition process.

CHAPTER 4: EXPERIMENTAL RESULTS

1. Introduction:

In this chapter, we present the experimental results of our research. In this section, we discuss the handwritten Arabic word segmentation system and the recognition system for handwritten Arabic letters segmented by deep learning approach. We used a convolutional neural network (CNN) architecture. The model was trained on a good dataset to be able to recognize well, as will be shown in this chapter. We experimented with several hyperparameters, including activation functions, optimizers, and kernel initializers, to improve the model performance.

2. Testing:

The purpose of the test is to see how effective is the word segmentation of our segmentation system and compare the outputs of the neural network on the test set.

The results of segmenting separate words were always excellent, as shown in the interface, and this is due to the fact that the space between the two letters does not always reach the threshold:

As for testing the model, Table 1 summarizes the results obtained after applying the proposed CNN model:

Table 5 Accuracy

	Loss value	Accuracy value
Training set	6.15%,	99.05%,
Test set	1.6%,	96.6%,

Some test experiments on our model:

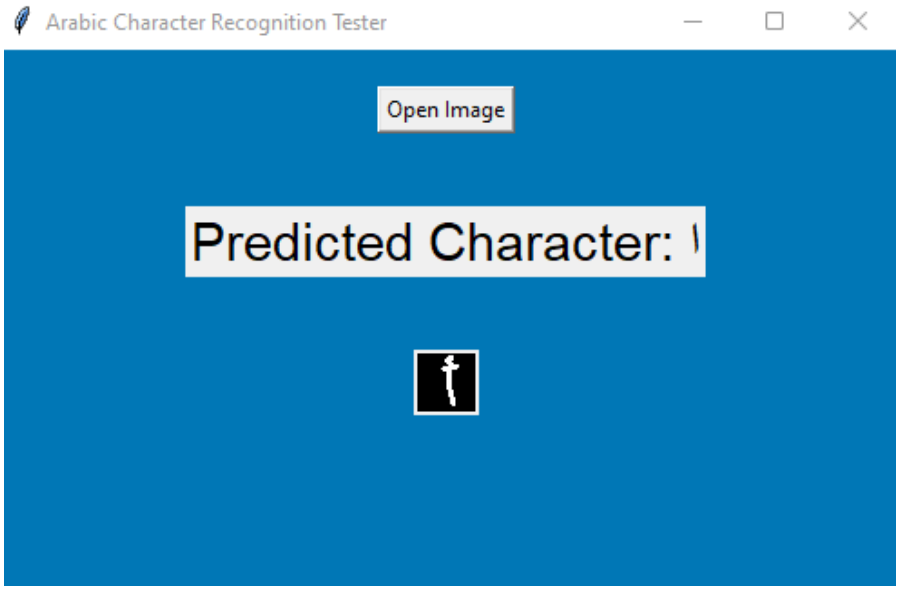

Figure 23 Predicted Character ا Real Character أ

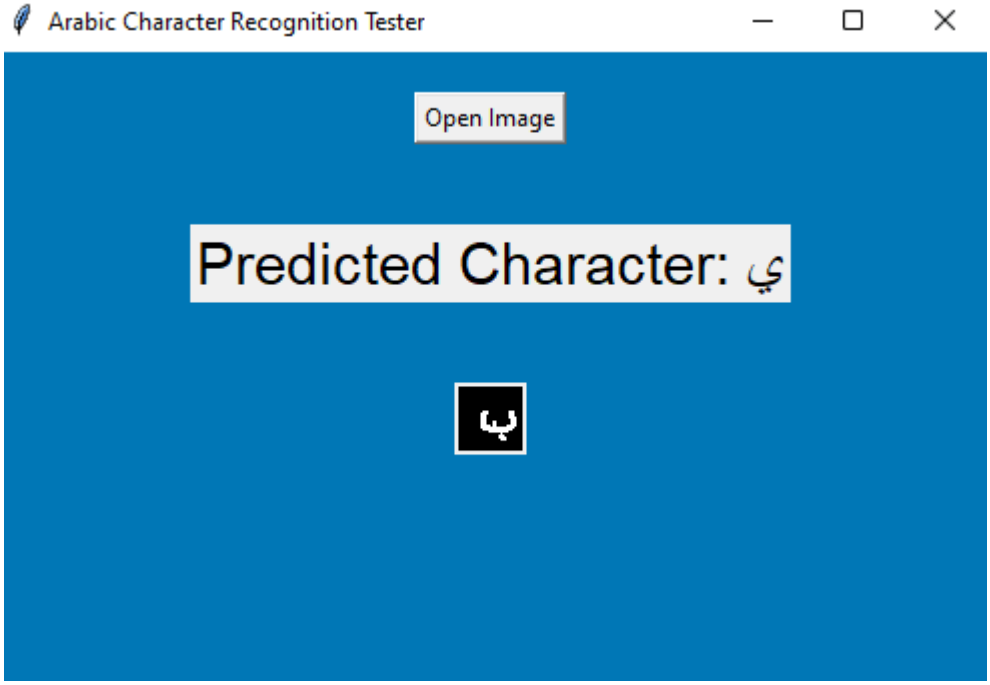

Figure 24 Predicted Character ي Real Characterب

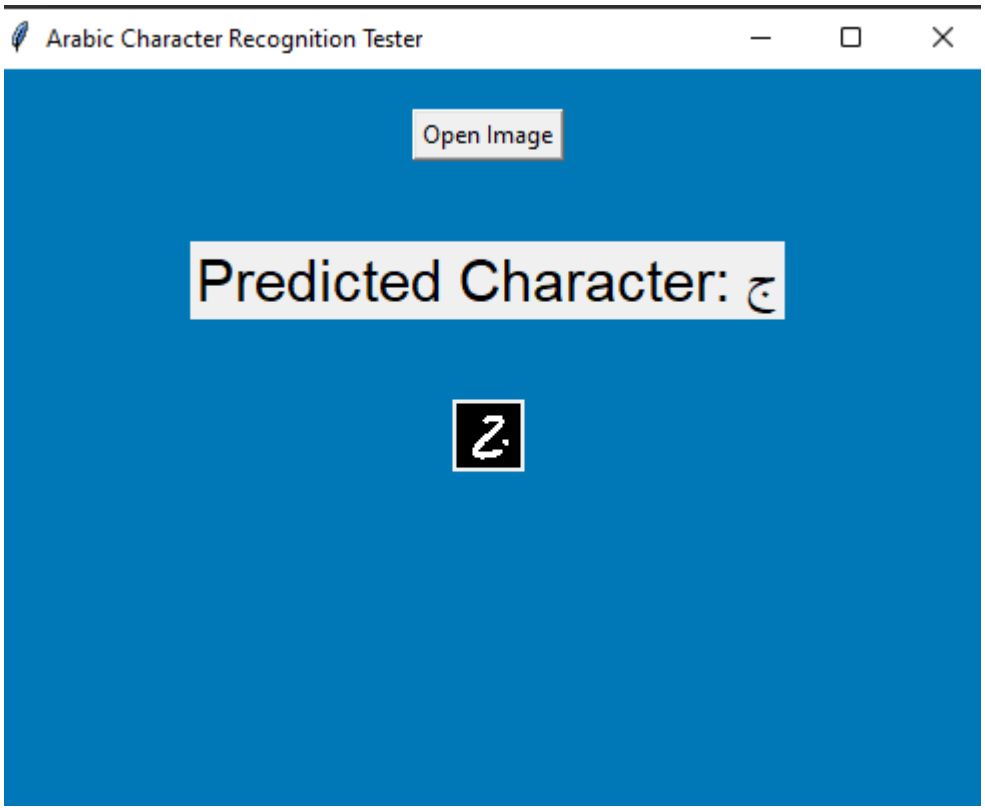

Figure 25 Predicted Character جReal Characterج

As shown in Figures 23, 24, and 25, the model recognizes many letters, but it sometimes makes mistakes in letters with very similar writing, such as the letter (ب) and the letter (ي), or the letter (ج)and the letter (ح).

3. Conclusion:

Convolutional neural networks (CNNs) are robust models that excel in character classification tasks, demonstrating impressive accuracy rates, especially in recognizing handwritten Arabic letters. They have been successful in achieving an accuracy rate as high as 99.05% for this specific task.

It's important to be cautious about using too many epochs when training a CNN because excessive epochs can lead to overfitting. Overfitting occurs when the model becomes overly specialized in the training data, essentially memorizing it rather than learning the underlying patterns. To mitigate this issue, it's essential to monitor the validation accuracy at each epoch (or even iteration) to determine if the model's performance is still improving or if it's starting to degrade.

Furthermore, increasing the number of convolutional layers in a CNN can indeed enhance the network's performance to some extent. This additional depth allows the model to capture more intricate features and patterns within the data. However, it's crucial to strike a balance because an excessively deep network may also lead to overfitting if not properly regularized.

In summary, CNNs are powerful tools for character classification, particularly for handwritten Arabic letters, but training parameters such as the number of epochs and the network's depth should be carefully managed to avoid overfitting and optimize performance.

Conclusion and Future Work

While this research began as a preliminary study in the field of handwritten Arabic word segmentation and handwritten Arabic letter recognition, it has made significant progress in the right direction. The results were promising, showing a commendable level of accuracy compared to previous research. Thus, the work we have done represents an important step forward and a valuable contribution to the understanding of the Arabic language. Also, by integrating segmentation and recognition research, this research can serve as a basic starting point for many future studies in the field of language analysis.

Undoubtedly, the issues of segmentation and handwriting recognition are of great importance due to their wide applications in various aspects of life, benefiting Arabic speakers and individuals interested in the Arabic language, regardless of their background. In addition, handwriting recognition programs are greatly enriching the scientific knowledge base in the Arab world. Moreover, this research holds great importance in the identification of ancient documents and manuscripts, allowing global access to these valuable resources through automated recognition of Arabic writing. This in turn allows for the exploration of the vast literary and scientific material present in the Arabic language, which would otherwise remain untapped without advanced educational models capable of recognizing the various ancient Arabic scripts.

The field of handwriting recognition is pivotal, given its diverse applications, and constantly requires investments to enhance the accuracy and relevance of Arabic language datasets. Object recognition relies primarily on machine learning techniques, with a recent focus on deep learning, a subfield that relies heavily on convolutional neural networks (CNNs). Accordingly, we have devised a system to recognize printed Arabic letters using deep learning technology. Our system features an efficiently implemented CNN for feature extraction and character classification.

In this study, we used convolutional neural networks to recognize handwritten Arabic letters. The network was trained using a dataset collected from 60 individuals between the ages of 19 and 40, comprising 28 distinct categories and a total of 16,800 images. Next, this dataset was divided into training (80%) and testing (20%) sets. The CNN model was trained using 13,440 images and tested using 3,360 images. Our observations indicate that increasing the number of convolutional layers as well as the number of training times enhances the discriminatory ability of the system.

An overview of artificial intelligence and its contributions to natural language processing is provided, focusing on the importance of the Arabic language globally. It also presented the research problem, identified the main obstacles that hinder the development of Arabic letter recognition systems, and discussed the research objectives and their importance. In addition, different types of letter recognition were examined.

In Chapter 1, we detail a dataset consisting of handwritten Arabic letters, totaling 16,800 images divided into 28 categories.

The second chapter presented a review of previous studies on letter recognition conducted between 2017 and 2023.

The third chapter dealt with the stages involved in building a segmentation system and a handwritten Arabic character recognition system, including image acquisition, pre-processing, feature extraction, and classification. The feature extraction and classification stages rely heavily on convolutional neural networks.

The fourth chapter presented the results of training the model and discussed the segmentation system, where both models achieved high accuracy and low loss. The importance of the number of layers and training cycles in influencing recognition accuracy was emphasized.

In conclusion, this research represents a significant contribution, providing insights into word segmentation and handwritten Arabic character recognition. Future prospects include creating a high-quality dataset that includes all positions of Arabic script letters to further enhance our model and potentially contribute to the interpretation of Arabic words. In addition, future work could include expanding the segmentation phase to include connected and discrete words, focusing on improving Arabic language databases for the research community, and incorporating various old and new Arabic fonts to expand the applicability of the project.

References

[1] Noack, R. (2015). The future of language. The Washington Post.

[2] Ella, S. What Is a Dataset? Definitive Guide. brightdata.com.

https://brightdata.com/blog/web-data/what-is-a-dataset

[3] Sydorenko, I. (2021). What is a Dataset in Machine Learning: Sources, Features, Analysis. labelyourdata. com.

[4] Rafiuddin, K. (2022).Importance of Datasets in Machine Learning and AI Research. datatobiz.com.

https://www.datatobiz.com/blog/datasets-in-machine-learning/

[5] Mirko, S. (2020). Split Your Dataset With scikit-learn's train_test_split(). realpython.com.

https://realpython.com/train-test-split-python-data/

[6] Joby, A. (2023). What is training data? how it’s used in machine learning.

[7] Jared Wilber & Brent Werness, The Importance of Data Splitting.

[8] Loey, M. (2019). Arabic handwritten characters dataset.

[9] Altwaijry, N., & Al-Turaiki, I. (2021). Arabic handwriting recognition system using convolutional neural network. *Neural Computing and Applications*, *33*(7), 2249-2261.

[10] El-Sawy, A., Loey, M., & El-Bakry, H. (2017). Arabic handwritten characters recognition using convolutional neural network. *WSEAS Transactions on Computer Research*, *5*(1), 11-19.

[11] Alwagdani, M. S., & Jaha, E. S. (2023). Deep Learning-Based Child Handwritten Arabic Character Recognition and Handwriting Discrimination. *Sensors*, *23*(15), 6774.

[12] Balaha, H. M., Ali, H. A., Youssef, E. K., Elsayed, A. E., Samak, R. A., Abdelhaleem, M. S., ... & Mohammed, M. M. (2021). Recognizing arabic handwritten characters using deep learning and genetic algorithms. *Multimedia Tools and Applications*, *80*, 32473-32509.

[13] Mudhsh, M., & Almodfer, R. (2017). Arabic handwritten alphanumeric character recognition using very deep neural network. *Information*, *8*(3), 105.

[14] Python. aws.amazon.com

https://aws.amazon.com/ar/what-is/python/

[15] Analytics Vidhya (2023).Top 10 Machine Learning Libraries You Should Know in 2023. analyticsvidhya.com.

https://www.analyticsvidhya.com/blog/2023/03/machine-learning-libraries/

[16] Rouse, M. (2019). What Does Matplotlib Mean. techopedia.com. https://www.techopedia.com/definition/33861/matplotlib

[17] Tkinter. en.wikipedia.org

https://en.wikipedia.org/wiki/Tkinter

[18] Loey, M. (2019). Arabic handwritten characters dataset.

[19] Abandah, G. A., & Khedher, M. Z. (2009). Analysis of handwritten Arabic letters using selected feature extraction techniques. *International Journal of Computer Processing of Languages*, *22*(01), 49-73.

Abstract:

Many languages have made significant advancements in the field of character recognition, including English, Chinese, Japanese, and French, with recognition rates reaching up to 100% in some cases. However, Arabic handwriting recognition faces lower recognition rates, primarily due to certain linguistic characteristics that make the recognition process more challenging, along with a shortage of high-quality available datasets.

Therefore, this memorandum was undertaken with the aim of developing a system for recognizing handwritten Arabic characters and a word segmentation system. The study began with an analysis of the Arabic language's structure, followed by an overview of deep neural network technology, which has proven its efficiency in achieving rapid and reliable recognition results. Finally, the obtained results were explained and interpreted.

Keywords: recognition of handwritten Arabic letters; segmentation of handwritten Arabic words; convolutional neural networks; processing; Feature extraction; classification;

Résumé :

De nombreuses langues ont fait d'importants progrès dans le domaine de la reconnaissance des caractères, notamment l'anglais, le chinois, le japonais et le français, avec des taux de reconnaissance atteignant jusqu'à 100% dans certains cas. Cependant, la reconnaissance de l'écriture manuscrite en arabe présente des taux de reconnaissance plus faibles, principalement en raison de certaines caractéristiques linguistiques qui rendent le processus de reconnaissance plus difficile, ainsi que d'une pénurie de jeux de données de haute qualité disponibles.

Par conséquent, cette note a été rédigée dans le but de développer un système de reconnaissance des caractères arabes écrits à la main et un système de segmentation des mots. L'étude a débuté par une analyse de la structure de la langue arabe, suivie d'une présentation de la technologie des réseaux neuronaux profonds, qui a fait ses preuves en termes d'efficacité pour obtenir des résultats de reconnaissance rapides et fiables. Enfin, les résultats obtenus ont été expliqués et interprétés.

Mots-clés : reconnaissance de lettres arabes manuscrites ; segmentation de mots arabes manuscrits ; réseaux de neurones convolutifs ; traitement ; Extraction de caractéristiques ; classification ;

الملخص:

تطورت العديد من اللغات بشكل كبير في مجال التعرف على الحروف، من بينها الإنجليزية، الصينية، اليابانية، والفرنسية. وقد وصلت نسب التعرف في بعض الحالات إلى 100%. ومع ذلك، يشهد التعرف على الكتابة العربية اليدوية نسب منخفضة، وهذا يرجع إلى بعض الخصائص اللغوية العربية التي تجعل عملية التعرف أكثر صعوبة، بالإضافة إلى نقص في جودة مجموعات البيانات المتاحة .

لذا، تم تنفيذ هذه المذكرة بهدف تطوير نظام للتعرف على الحروف العربية المكتوبة بخط اليد ونظام لتجزئة الكلمات. تم البدء بدراسة هيكل اللغة العربية، تلتها استعراض لتقنية الشبكات العصبية العميقة التي أثبتت فعاليتها في التعرف السريع وتحقيق نتائج موثوقة. وفي الختام، تم شرح وتفسير النتائج التي تم الوصول إليها.

الكلمات المفتاحية: التعرف على الحروف العربية المكتوبة بخط اليد؛ تجزئة الكلمات العربية المكتوبة بخط اليد؛ الشبكات العصبية الالتفافية؛ المعالجة؛ استخراج السمات؛ التصنيف؛