electronics

MDPI

Article

Toward Building a Domain-Based Dataset for Arabic Handwritten Text Recognition

Khawlah Alhefdhi [1,*], Abdulmalik Alsalman [1] and Safi Faizullah [2,*]

[1] Computer Science Department, King Saud University, Riyadh 11362, Saudi Arabia; salman@ksu.edu.sa
[2] Department of Computer Science, Islamic University, Madinah 42351, Saudi Arabia
* Correspondence: khawlahalhefdhi.ksu@gmail.com (K.A.); safi@iu.edu.sa (S.F.)

Abstract: The problem of automatic recognition of handwritten text has recently been widely discussed in the research community. Handwritten text recognition is considered a challenging task for cursive scripts, such as Arabic-language scripts, due to their complex properties. Although the demand for automatic text recognition is growing, especially to assist in digitizing archival documents, limited datasets are available for Arabic handwritten text compared to other languages. In this paper, we present novel work on building the Real Estate and Judicial Documents dataset (REJD dataset), which aims to facilitate the recognition of Arabic text in millions of archived documents. This paper also discusses the use of Optical Character Recognition and deep learning techniques, aiming to serve as the initial version in a series of experiments and enhancements designed to achieve optimal results.

Keywords: OCR; Arabic handwritten recognition; optical character recognition; Arabic handwritten dataset; REJD dataset

1. Introduction

Automatic recognition of the textual content of documents is profitable, bringing up numerous new business opportunities [1]. However, the automatic recognition process is a complex task due to the variety of document characteristics and the nature of the text language. Recognizing Arabic script is a complex task for several reasons, including the fact that Arabic has a morphologically and cursively rich script [2].

Automated methods for Arabic text recognition still require more research and effort compared to the efforts achieved for English, Latin, and Chinese text recognition [3,4]. The more challenging task is recognizing handwritten text, and the need for digitization is increasing due to the expansion of handwritten Arabic archives in libraries, data centers, historical institutions, and workplaces. For example, governments need to digitize essential documents such as notary records and title deeds, making them easier to keep and manage [5].

Optical Character Recognition (OCR) is the technique of detecting and recognizing text in images and transforming it into editable text. The construction of handwritten text recognition models, a special case of OCR models [4], is now an ongoing task in digital humanities initiatives [6], which is attributable to the wide advancement and promising results obtained from deep learning approaches. Most available Arabic OCR techniques have been evaluated in controlled environments with specific constraints, such as datasets containing isolated characters or words in high-quality images.

check for updates

Academic Editor: Ricardo Martins

Received: 29 April 2025
Revised: 19 May 2025
Accepted: 28 May 2025
Published: 17 June 2025

Citation: Alhefdhi, K.; Alsalman, A.; Faizullah, S. Toward Building a Domain-Based Dataset for Arabic Handwritten Text Recognition. *Electronics* **2025**, *14*, 2461. https://doi.org/10.3390/electronics14122461

There is an increasing demand for having a wide variety of Arabic datasets, as the size and quality of the datasets affect the recognition results. The large amount and high quality of the images lead to higher accuracies.

Given the current state of the field, this research aims to discuss these techniques in parallel with building the dataset required to achieve the goal of text recognition of archived documents. This research's main contribution is the creation of a large dataset that will enrich the Artificial Intelligence (AI) applications that support Arabic-language projects, especially for real estate and judicial documents. To the best of our knowledge, this work is the first one in this area.

This paper is structured as follows: Section 2 provides a review of the literature and related works; Section 3 describes the steps of building the Real Estate and Judicial Documents dataset (REJD dataset); Section 4 presents a discussion of the work and results; and finally, Section 5 presents the conclusion and future work.

2. Literature Review

Research in the field of Arabic handwritten text recognition has been becoming increasingly diverse in recent years. In this section, a review of the papers and work related to the paper's topic is presented. First, existing datasets are surveyed, after which the judicial and legal documentation is reviewed.

2.1. Review of Existing Datasets

In this field, there are many datasets for the language's script. Datasets can be categorized based on different criteria, including:

- Language: one language or multiple languages.
- Recognition Level: characters, digits, words, lines, or paragraphs.
- Task: text recognition, writer identification, and so on.
- Script Type: printed, handwritten, or both.
- Recognition Type: online or offline.

IFN/ENIT [1] is one of the most common datasets for handwritten Arabic text. This dataset was developed by the Institute for Communications Technology/Ecole Nationale d'Ingénieur de Tunis. It is limited to 946 words of Tunisian town names, written by 411 writers. The number of images in this dataset is 26,459.

Another notable dataset is Muharaf [2], which is a dataset of manuscripts of handwritten Arabic consisting of over 1600 images. The nature of the manuscripts is historical documents, including letters, poems, and legal records. Each image is associated with coordinates of its text regions and lines, including the headings, main text, and other page elements. Within the dataset, some images are public, and others are restricted.

Another dataset, known as Hijja [5], is a collection of handwritten Arabic letters from children aged 7–12 years. The total number of images is 47,434 characters, written by 591 participants. The Hijja dataset is organized into 29 folders, representing 28 Arabic alphabets in addition to "hamza". Also, this dataset contains subfolders for different forms of each letter.

Moj-DB is a dataset of Arabic historical handwriting that contains 560,000 sub-words extracted from 64 pages of 129 pages selected from 10 ancient Arabic books. Zoizou et al. [4] proposed an approach for extracting sub-words from the text lines.

Another dataset presents handwritten Arabic characters, called AHAWP [6], which contains 65 different Arabic alphabet forms, 10 words, and 3 paragraphs. The dataset was gathered from 82 users, each of whom wrote each alphabet and word 10 times. Therefore, the dataset contains a total of 53,199 alphabet images, 8144-word images, and 241 paragraph images. This dataset can be used for OCR of handwritten Arabic and writer identification.

The HMBD dataset [7] is a dataset of Arabic handwritten characters that captures different positions of Arabic handwritten characters, including letters and digits. It contains 54,115 images of 115 classes in the dataset. The dataset was gathered from 125 volunteers. This dataset can be used to develop deep learning systems for Arabic handwritten text recognition.

Another dataset [8] is extracted from ancient Arabic Text, which provides both the image and textual ground truth for a collection of ancient Arabic manuscripts.

Table 1 summarizes the most common datasets in the scope of Arabic handwritten text recognition.

Table 1. Arabic handwritten text datasets.

Dataset	Reference	Data Level	Data Size (# Images)	Availability
IFN/ENIT (2002)	M. Pechwitz et al. [1]	Words	26,549	Public/free
KHATT (2012)	S. A. Mahmoud et al. [3]	Paragraphs	4000	Public/free
Hijja (2021)	N. Altwaijry et al. [5]	Characters (alphabets)	47,434	Public/free
HMBD (2021)	H. M. Balaha et al. [7]	Numbers, Characters	54,115	Public
MOJ-DB (2022)	A. Zoizou et al. [4]	Sub-words	560,000	Public/free/upon request
AHAWP (2022)	M. A. Khan [6]	Words, alphabets, paragraphs	61,584	Public/free
Muharaf (2025)	Saeed et al. [2]	Text lines	1644	Partially public

2.2. Review of Research for Legal Documents

Generally, it is helpful to explore the latest studies and research efforts for Arabic and non-Arabic languages and scripts in the scope of legal and judicial documents or documents that have a similar nature, such as archives/historical manuscripts.

In [9], the authors discussed a strategy for collecting relevant metadata from legal documents, which are particularly difficult to process due to their length and varied structure and style. Practically, the proposed approach employs the most recent models while simultaneously preserving information about the structure and formatting of the page (i.e., visual cues). The key to this strategy is the usage of characteristics from OCR above and beyond the document's features. It evaluated the proposed approach using the Contract Understanding Atticus Dataset (CUAD), which includes 510 English-language commercial legal contracts with rich expert annotations. The proposed method outperformed previous models (expert rules and a large-scale pre-trained language model, DeBERTa [10]) on this dataset for significant contract-understanding tasks.

With the aim of building monolingual and parallel legal corpora of Arabic and English countries' constitutions, El-Farahaty et al. [11] discussed this aim to open the path for more research into Arabic legal translation and teaching. This field is still under-researched globally. A key challenge in this area is the lack of effective Arabic OCR, so the paper reported on using multiple free online OCR tools (e.g., Sotoor OCR) to manage recognition difficulties relating to the efficiency, accuracy, and encoding issues associated with Arabic. In the same context, Gupta et al. [12] discussed the creation and analysis of an international corpus of privacy laws with the challenges of using available OCR tools.

The 206 System [13] integrated many AI techniques to exploit and analyze accumulated criminal cases and judicial information resources, including OCR, Natural Language Processing (NLP), speech recognition, and judicial entity identification. It is a code name related to the Shanghai High People's Court in China, and it adopted OCR technology and

a deep neural network to train around 15,000 case files. It was capable of fully recognizing all types of printed evidence and some types of handwritten text (e.g., signatures), as well as extracting and confirming relevant information based on predefined rules.

Additionally, as law courts spend substantial time reading and assessing legal case types, there is a need for a multi-labeled classification method. One such method has been proposed by Qiu et al. [14], referred to as MLTCCN. In their research, the authors used OCR as the first step to obtain text data from images taken from legal documents in regional courts in China. The authors also added tools for word segmentation and a vocabulary generator used for word mapping to obtain the final required classification. The results indicated efficiency in detecting errors and deviations in similar instances across district courts.

Table 2 summarizes recent research works from the literature, sorted according to publishing year from oldest to newest.

Table 2. Summary of recent related work.

Reference	Year	Script Language	Document Type	Main Task	Methodology
[15]	2018	Japanese	Historical manuscripts	Reprint support system for reading manuscripts	Constraint solving web service
[16]	2018	English script (Brazil)	Lawsuit cases in courts	Document classification	BDLSTM
[17]	2019	Multiple languages	Many datasets	Gender prediction	Feature extraction and classification methods (GLBP and HOG)
[18]	2020	Chinese	Legal documents	Document multilabel classification	MLTCNN
[19]	2021	English script (Spanish)	Colonial notary records	Content extraction	Deep learning pre-trained model evaluation
[20]	2022	Arabic script (Ottoman/Turkish)	19′th–20′th-century periodicals	Automated transcription	Deep learning method of HTR models
[21]	2022	Arabic script (Maghrebi)	Old manuscripts	Text recognition	Word-based neural approach
[22]	2023	Devanagari script (Nepali)	Students' handwritten scripts	Text recognition	CRNN
[23]	2023	Bengali	Six domains, including public domain government documents and property deeds	Document layout analysis (DLA)	Building a large multi-domain DLA dataset
[8]	2024	Arabic	Historical book and page-level transcriptions	Text recognition	Deep learning OCR modeling and testing

3. REJD Dataset

The data in hand is a collection of real estate archival manuscripts. Notably, there are different types of such documents, including several layouts and different writing styles. In this paper, we propose a dataset built from one type of these manuscripts. The writing script is a combination of printed and handwritten styles. Additionally, the document layout is a form-shape, where the printed parts are common for all documents, but on the other hand, the handwritten parts are completed by different writers.

3.1. Data Specifications

A substantial number of archival documents exist; furthermore, the task of handling them is chosen to occur in phases according to several criteria such as a document's location (i.e., its physical storage location), text types (fully printed, fully handwritten, or hybrid), document layout, time period, and so on. In this paper, we focus on the first version of our dataset, which is built from approximately 110,000 pages of archival documents obtained from the organization branches located in the same city. The dimensions of the scanned images are 3504 × 4960 pixels, with a resolution of 300 dpi, and they are stored in TIF file format. The nature of text in these documents is hybrid and form-shaped, wherein the fixed text is printed and the free text is handwritten. Figure 1 shows a sample image of this type of document, with some parts of the image redacted for privacy reasons.

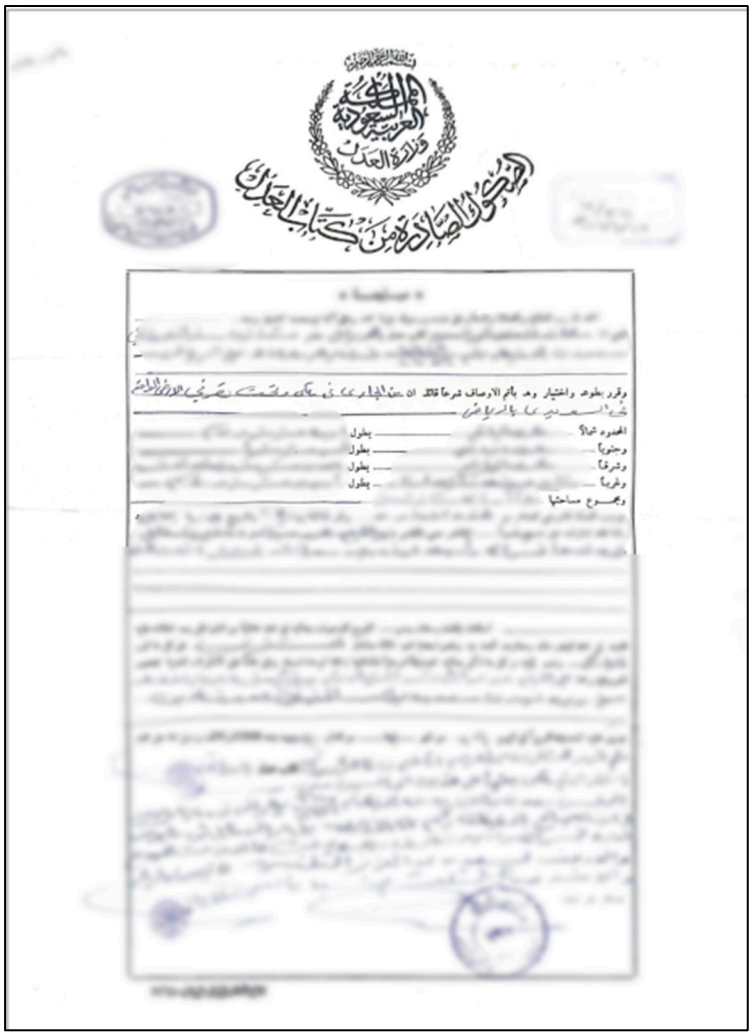

Figure 1. Sample of an archival document (https://www.moj.gov.sa/Documents/RealetateDeedUpdate.pdf, accessed on 26 November 2024).

3.2. Data Preparation and Segmentation

Initially, we aimed to recognize only specific text, including the writer's name, buyer's name, and deed date. Fortunately, we had some metadata about the documents in hand, so we performed data analysis to derive valuable insights, such as the range of years of deed dates and the names of the top 10 writers. As a result, the data built for this version of the dataset contained digits (for deed numbers and dates) and words (for names and commonly used terms). We constructed the dataset content by adopting two tracks, as discussed in the following subsections.

3.2.1. Using OCR Software Tool

By analyzing the document layout, we identified areas of interest that were expected to include the text required for optical recognition. Using Python scripts and EasyOCR, an OCR tool [24], we calculated the area of the required text, cropped them to form snippets of text, and then saved them as JPG files along with their corresponding JSON files (see Figure 2). Figure 3 shows a sample of an area to be cropped. All images of the snippets were stored as JPG files and automatically named in a format describing their labels and original image names.


```
  "c1": "[np.int32(2988), np.int32(589)]",
  "c2": "[np.int32(3118), np.int32(589)]",
  "c3": "[np.int32(3118), np.int32(649)]",
  "c4": "[np.int32(2988), np.int32(649)]",
  "text": "التسجيل",
  "threshold": "0.9333623169110286"
},
{
  "c1": "[np.int32(2051), np.int32(613)]",
  "c2": "[np.int32(2226), np.int32(613)]",
  "c3": "[np.int32(2226), np.int32(734)]",
  "c4": "[np.int32(2051), np.int32(734)]",
  "text": "افراغ",
  "threshold": "0.8894613873523386"
},
{
  "c1": "[np.int32(2296), np.int32(600)]",
  "c2": "[np.int32(2504), np.int32(600)]",
  "c3": "[np.int32(2504), np.int32(716)]",
  "c4": "[np.int32(2296), np.int32(716)]",
  "text": "مبايعة",
  "threshold": "0.881200150055619"
},
{
  "c1": "[np.int32(3010), np.int32(658)]",
  "c2": "[np.int32(3115), np.int32(658)]",
  "c3": "[np.int32(3115), np.int32(713)]",
  "c4": "[np.int32(3010), np.int32(713)]",
  "text": "التسلم",
  "threshold": "0.874795218307114"
},
{
  "c1": "[np.int32(1980), np.int32(788)]",
  "c2": "[np.int32(2851), np.int32(788)]",
  "c3": "[np.int32(2851), np.int32(881)]",
  "c4": "[np.int32(1980), np.int32(881)]",
  "text": "[illegible]",
  "threshold": "0.08947451047683289"
},
{
  "c1": "[np.int32(2874), np.int32(797)]",
  "c2": "[np.int32(2977), np.int32(797)]",
  "c3": "[np.int32(2977), np.int32(857)]",
  "c4": "[np.int32(2874), np.int32(857)]",
  "text": "لدي انا",
  "threshold": "0.693153393022375"
},
```


Figure 2. Excerpt of a JSON file generated using EasyOCR.

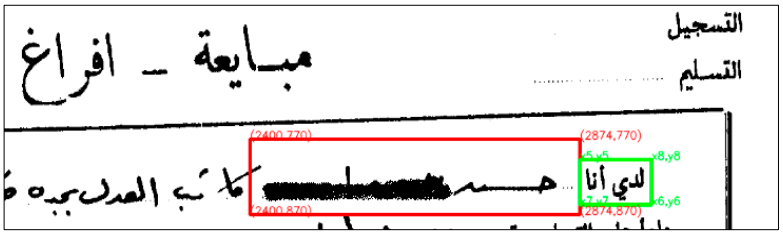

Figure 3. Coordinates of the recognized word "لدي أنا" and the text beside it to be cropped and recognized.

Since the nature of the data used in this research is sensitive, the work was restricted, and it was decided to be kept private. Therefore, we constructed an internal interface for annotation purposes, where the annotators were provided with credentials to log into the tool and explore the snippets for the annotation process. The total number of annotators was nine adult Arabic people. The tool mainly enabled the annotators to choose the correct text from the list of predefined options, rather than writing it, except in some cases. If the text was unclear in one snippet image, the annotator had the option to skip it. Once the annotator selected a button of the displayed option of the annotated text, the snippet image was saved along with its label, and the next random snippet was displayed on the screen for the next labeling. Figure 4 shows screenshots of how the annotation process was achieved.

Figure 4. Annotation process windows: (**a**) days; (**b**) months; (**c**) years; (**d**) writers; (**e**) buyer first name; (**f**) buyer second name.

The list of days consisted of 30 numbers ranging from 1 to 30, representing the days in a month as per the Hijri calendar. Therefore, we displayed them in a 6×5 matrix for easy selection. Also, a 4×3 matrix of 12 months was provided for the month annotation. For years, we used a range of 15 years, so we displayed 15 options in a 5×3 matrix, according to the Islamic calendar.

On the other hand, for the writer's name, we provided a list of top writers and enabled the annotators to either choose one of the displayed names, write the seen name if it was not in the list, or simply skip the current image to jump to the next one. Also, for the buyer's name, we facilitated the annotation by providing two options: the first name and the second name. For that, we listed the most common names found in the documents, providing annotators with the ability to choose other names from a dropdown menu. It is noteworthy that this process resulted in obtaining the ground truth of the text in the images.

Notably, the annotators were not fully dedicated to this task, and so the process of annotation was somehow slow. The number of snippets totaled approximately 330,000 images, and only around 57,260 snippets were annotated, comprising approximately 17% of the total snippets.

3.2.2. Pre-Processing of the Crops

We performed pre-processing procedures for the annotated images, considering each one as a text line. These procedures were achieved using Visual Studio IDE and the Python programming language. OpenCV was utilized as an essential library for image processing, offering tools for both enhancement and processing [25].

First, we changed the color to a grayscale. Then, a threshold was applied to obtain binary images. After that, we applied erosion to reduce the dots in the images and dilation and to identify large areas of interest (ROIs). Then, we identified contours based on not-touching borders and meeting the minimum width and height (specified as 20 × 20). Finally, through trial and error, we unified the size of each digit image as 80 × 80 and each word image as 160 × 80 using the padding feature [26].

The initial goal was to have a model with the ability to detect at least some of the known words, in which each of these predefined words represents a class as follows:

DigitClasses = [“0”,”1”,”2”,”3”,”4”,”5”,”6”,”7”,”8”,”9”]

NameClasses = [“Muhammad”,”Ali”,”Abd”,”Allah”,”/”]

Tables 3 and 4, respectively, show the lists of digits and main words used in the experiments to build the REJD dataset. It is worth mentioning that the corpus of the required words will increase depending on the type of documents themselves. Figures 5 and 6 show sample images from the REJD dataset.

Table 3. Digit images in the REJD dataset.

Digits in English Script	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Total**
Digits in Arabic Script	٠	١	٢	٣	٤	٥	٦	٧	٨	٩	
# HMBD Digits	480	241	379	459	200	464	489	474	488	474	3668
# REJD Digits	0	33	105	119	277	51	73	36	45	64	803
Total	480	274	484	578	477	515	562	510	533	538	4471

Table 4. Word/special character images in the REJD dataset.

Word/Special Character	محمد	علي	عبد	لله	/	**Total**
Count	182	182	182	182	44	772

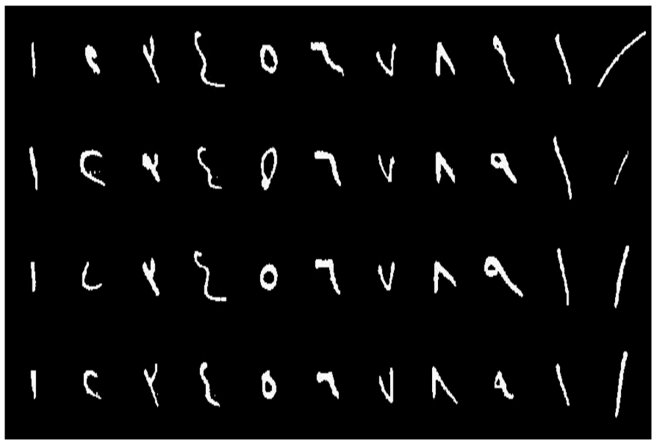

Figure 5. Samples of digits and special characters in the REJD dataset.

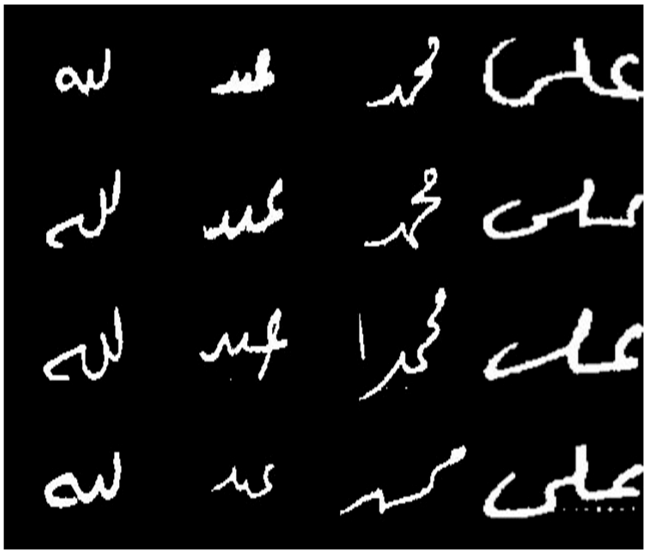

Figure 6. Samples of words in the REJD dataset.

4. Discussion

For training processes, we worked on a server with the following specifications: Intel Xeon Platinum 8163 processor, 2.50 GHz clock speed, 368 GB RAM, and 4 Tesla V100-SXM2-32GB GPU (total: 128 GB). We randomly started by using the Residual Network (ResNet) model, which is one of the popular CNN architectures that performs remarkably well due to its deep architecture and robust feature extraction capabilities [27]. This model was trained with different layers on millions of images from the ImageNet dataset, which serves as a benchmark for object localization algorithms [28]. The ResNet architecture is mainly based on the use of skip connections to reduce the vanishing gradient problem, leading to enhanced training on deeper networks [29]. Also, we found that ResNet and its variants, including ResNet-18 and ResNet-50, presented good results in the context of handwritten text recognition. Table 5 shows the main results obtained for our dataset, along with other related work.

Table 5. Training with ResNet models.

Related Work	Model	Dataset	Type	Accuracy
[30] 2022	ResNet-50	AIA9K	Alphabets	92.37%
		AHCD	Alphabets	98.39%
		Hijja	Alphabets	91.64%
[31] 2023	ResNet-18	AHCD	Alphabets	97.26%
[32] 2021	ResNet-34	MADBase	Digits	99.6%
This work	ResNet-50	HMBD	Digits	92%
		REJD + HMBD	Digits	98%
		REJD	Words	96.3%

Generally, we decided not to be limited to one model for all classes. Many models might lead to better results, as different classes may need different settings. The criteria for selecting the model included the complexity of the class pattern, scalability of the data, and level of confidence needed for the data in hand. However, for the current version of our dataset, we only applied ResNet-50. This model has the same concept as ResNet but is 50 layers deep, as shown in Figure 7.

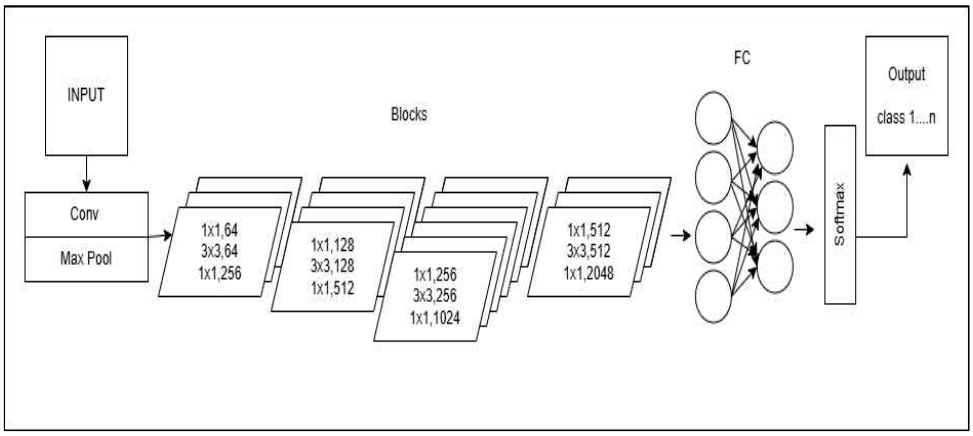

Figure 7. ResNet-50 architecture [30].

Two experiments were performed: one for the digits and another for the words and special characters. Model training was accomplished by assigning 10 epochs, with a batch size equal to 32. The dataset was split into three parts: training (70%), validation (20%), and testing (10%). As mentioned in Section 3.2, some pre-processing of the data was achieved, including transforming and resizing the images. Additionally, imbalanced data of some classes was managed through class weighting. We also used the HMBD dataset [7] for the purpose of data augmentation (Table 3), thereby increasing the diversity of data and the efficiency of the model.

The details of the training processes for the digits and words are shown in Tables 6 and 7, respectively. Figures 8 and 9 can be useful for the visualization of epoch/accuracy and epoch/loss graphs. Additionally, the confusion matrix of the classes that shows the performance of actual vs. predicted values is demonstrated in Figures 10 and 11.

Table 6. Digits training performance.

Training Metrics							
Epoch	Loss	Accuracy					
1	64.01%	80.20%					
2	16.24%	95.58%					
3	15.24%	95.84%					
4	12.12%	96.39%					
5	9.66%	97.37%					
6	5.57%	98.55%					
7	7.56%	97.92%					
8	8.81%	98.09%					
9	14.38%	96.16%					
10	9.27%	97.34%					
Validation Metrics				Test Metrics			
Accuracy	Error Rate	F1 Score	Precision	Accuracy	Error Rate	F1 Score	Precision
97.48%	2.52%	97.48%	97.52%	96.81%	3.19%	96.84%	96.99%
Class	Accuracy			Class	Accuracy		
0	100.00%			0	97.96%		
1	90.91%			1	96.43%		
2	95.88%			2	93.88%		
3	98.28%			3	94.83%		
4	97.92%			4	95.83%		

Table 6. *Cont.*

Training Metrics			
5	97.09%	5	98.08%
6	96.43%	6	96.49%
7	97.06%	7	98.08%
8	99.06%	8	98.15%
9	99.07%	9	98.15%

Table 7. Words training performance.

Training Metrics							
Epoch	Loss	Accuracy					
1	86.28%	67.68%					
2	49.11%	84.92%					
3	27.10%	90.66%					
4	20.11%	95.15%					
5	22.87%	94.43%					
6	10.55%	96.59%					
7	13.19%	96.05%					
8	9.75%	96.41%					
9	6.08%	97.85%					
10	6.65%	97.85%					
Validation Metrics				Test Metrics			
Accuracy	Error Rate	F1 Score	Precision	Accuracy	Error Rate	F1 Score	Precision
93.46%	6.54%	93.58%	94.32%	96.30%	3.70%	96.32%	96.53%
Class	Accuracy			Class	Accuracy		
1	97.22%			1	100.00%		
2	94.44%			2	94.74%		
3	86.11%			3	94.74%		
4	94.44%			4	94.74%		
5	100.00%			5	100.00%		

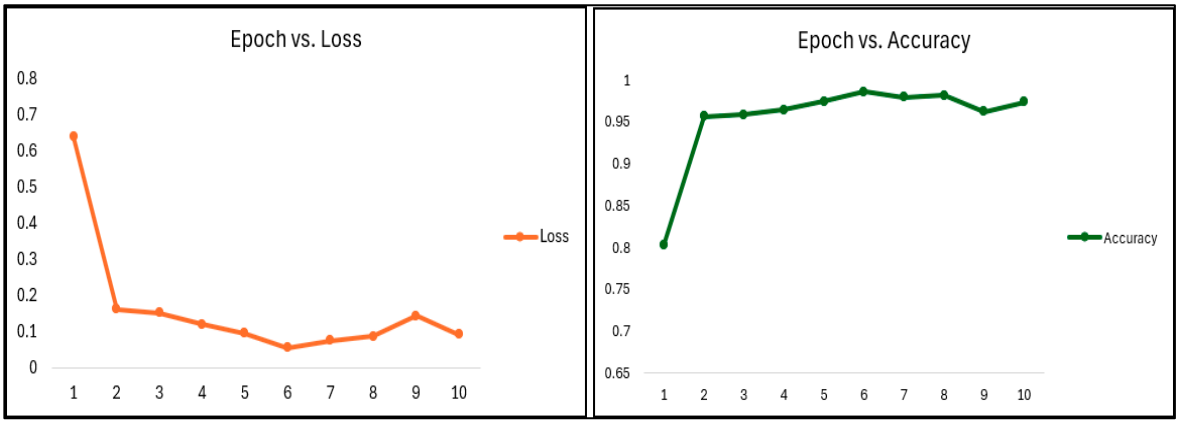

Figure 8. Epoch/accuracy and epoch/loss (digits).

The accuracy in this model is based on the number of correctly classified images compared to the total number of images. The model uses cross-entropy loss as a loss function, as it is suitable for the tasks of multi-class classification. This loss function calculates the difference in the probabilities between the true and the predicted class labels [33].

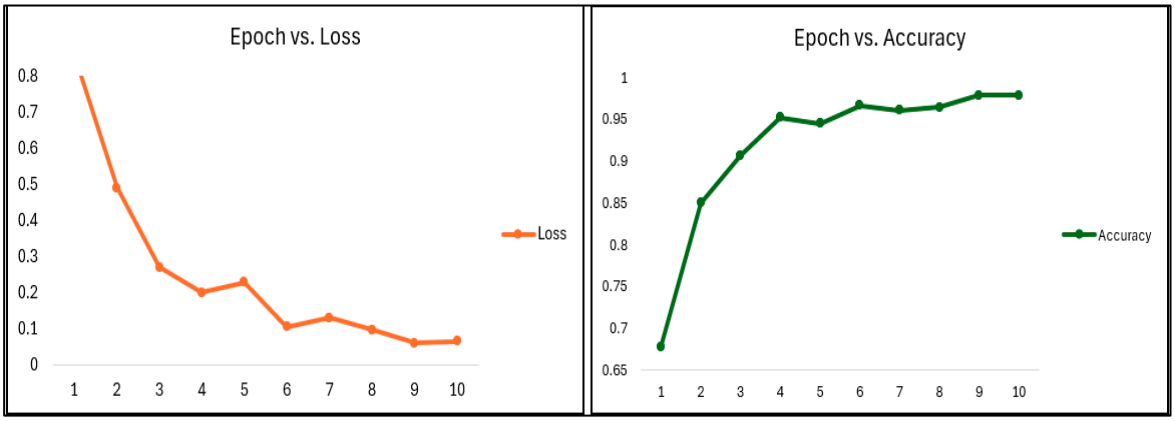

Figure 9. Epoch/accuracy and epoch/loss (words).

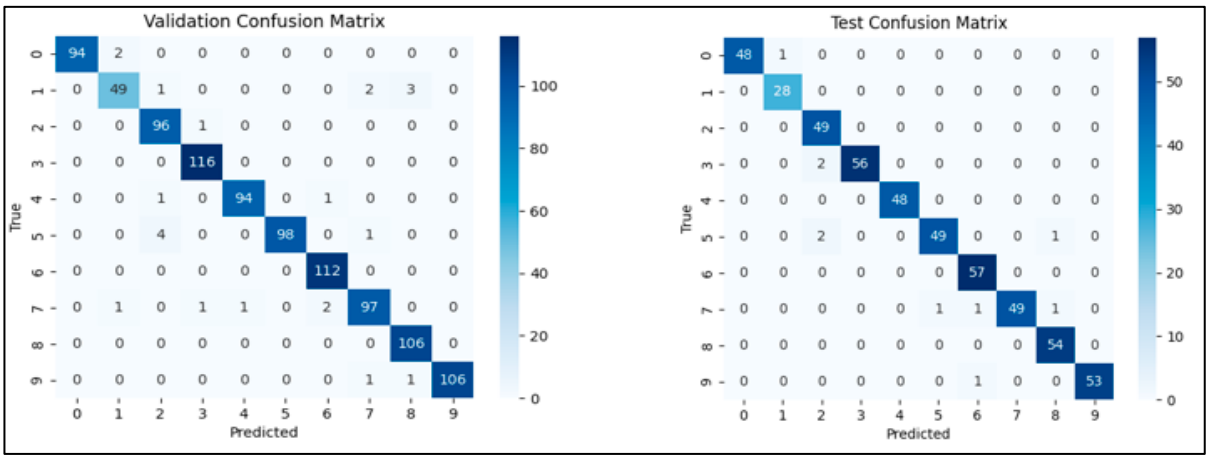

Figure 10. Confusion matrix (digits).

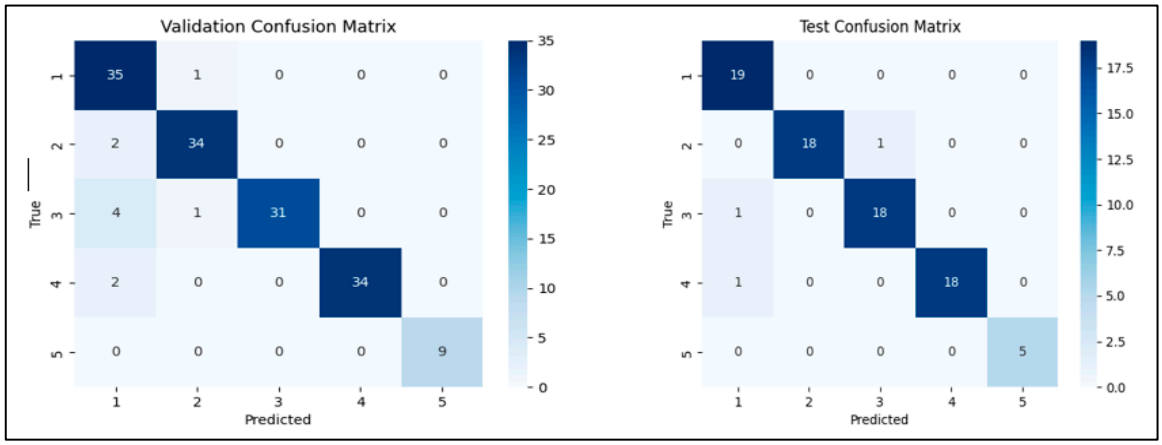

Figure 11. Confusion matrix (words).

Additionally, the model uses the Adam optimizer with a learning rate of 0.001. This optimizer combines the advantages of adaptive learning rates and momentum [33]. The model shows strong performance with high accuracy and a low error rate during training, validation, and testing, which indicates that the model is well-trained and generalizes effectively. Most classes achieved good accuracy (more than 95%). Comparing the results of digit classes among each other, digits "3" and "6" showed the best results. On the other hand, there were slight challenges with some classes, such as class 2, which represents the Arabic digit of the number "2", and class 3, which represents the Arabic digit of the number "3". For the words model,

classes of "Muhammad", "Ali", and "Abd" showed slight challenges, indicating the need for more focused training data or additional augmentations.

Considering the challenges in recognizing names with multiple parts, the dataset can be further augmented to improve the recognition of such names. For example, there are publicly available fonts that simulate handwritten styles, which can be used to produce texts leveraged for data augmentation to enhance variability in training datasets. Incorporating such fonts can help improve model generalization by exposing it to diverse writing patterns. Also, the integration of pre-trained Arabic NER (named entity recognition) models to refine segmentation and classification could be useful in future research.

In general, good results were obtained for the recognition of Arabic digits and words. However, there is still a need for further experiments, as the dataset will include more data and classes, which are intended to be discussed in future work. Also, some issues found during the experiments need to be addressed, such as names that are composed of two or more parts like "صالح", "إبراهيم", and the composition of digits and words like "٥/هـ ١٤٣١".

5. Conclusion and Future Work

To enhance the recognition process of archival documents, the Arabic script requires greater research efforts. This paper presents the first version of a proposed dataset, containing approximately 1575 enhanced images of Arabic digits and names, as a trial to enrich the data needed for recognition processes. Through image annotations and preprocessing, applying machine and deep learning techniques, and reviewing and comparing with the related works, this research has pursued important pillars that help toward building a useful dataset needed for the purpose of Arabic handwritten text recognition. For the recognition task, ResNet-50 showed good recognition accuracy, with 96.81% obtained for digits and 96.30% obtained for words. In future work, we aim to enhance the quantity and quality of the REJD dataset as well as enhance recognition accuracy for all classes. Also, accelerating the annotation process while maintaining high quality is a challenge, but several strategies can be implemented in future work to enhance efficiency, including batch processing, machine learning assistant annotation tools, and collaborative annotation platforms [34,35].

In the context of legal document recognition, the REJD dataset could be integrated with other legal-related datasets to enhance diversity, allowing for a broader representation of variations. Through style transfer techniques, dataset features can be blended, leading to improved generalization and more robust model performance. Consequently, this will enhance its practical applications.

Author Contributions: Conceptualization: K.A., A.A. and S.F.; data curation, K.A.; formal analysis, K.A.; investigation, K.A. and S.F.; methodology, K.A., A.A. and S.F.; project administration, K.A. and A.A.; resources, K.A.; software, K.A.; supervision, A.A. and S.F.; validation, K.A.; visualization, K.A.; writing—original draft, K.A.; writing—review and editing, K.A. and A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Publicly available data sources were used for this work, and restrictions were applied to the created dataset.

Acknowledgments: AI Team in the Ministry of Justice.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Pechwitz, M.; Maddouri, S.S.; Märgner, V.; Ellouze, N.; Amiri, H. IFN/ENIT-database of handwritten Arabic words. In Proceedings of the Francophone International Conference on Writing and Document, CIFED'02, Hammamet, Tunisia, 20–23 October 2002; Volume 2.
2. Saeed, M.; Chan, A.; Mijar, A.; Habchi, G.; Younes, C.; Wong, C.W.; Khater, A. Muharaf: Manuscripts of handwritten Arabic dataset for cursive text recognition. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 58525–58538.
3. Mahmoud, S.A.; Ahmad, I.; Al-Khatib, W.G.; Alshayeb, M.; Parvez, M.T.; Märgner, V.; Fink, G.A. KHATT: An open Arabic offline handwritten text database. *Pattern Recognit.* **2014**, *47*, 1096–1112. [CrossRef]
4. Zoizou, A.; Zarghili, A.; Chaker, I. MOJ-DB: A new database of Arabic historical handwriting and a novel approach for subwords extraction. *Pattern Recognit. Lett.* **2022**, *159*, 54–60. [CrossRef]
5. Altwaijry, N.; Al-Turaiki, I. Arabic handwriting recognition system using convolutional neural network. *Neural Comput. Appl.* **2021**, *33*, 2249–2261. [CrossRef]
6. Khan, M.A. Arabic handwritten alphabets, words and paragraphs per user (AHAWP) dataset. *Data Brief* **2022**, *41*, 107947. [CrossRef] [PubMed]
7. Balaha, H.M.; Ali, H.A.; Saraya, M.; Badawy, M. A new Arabic handwritten character recognition deep learning system (AHCR-DLS). *Neural Comput. Appl.* **2021**, *33*, 6325–6367. [CrossRef]
8. Najam, R.; Faizullah, S. A scarce dataset for ancient Arabic handwritten text recognition. *Data Brief* **2024**, *56*, 110813. [CrossRef]
9. Hegel, A.; Shah, M.; Peaslee, G.; Roof, B.; Elwany, E. The law of large documents: Understanding the structure of legal contracts using visual cues. *arXiv* **2021**, arXiv:2107.08128.
10. Hendrycks, D.; Burns, C.; Chen, A.; Ball, S. CUAD: An expert-annotated NLP dataset for legal contract review. *arXiv* **2021**, arXiv:2103.06268.
11. El-Farahaty, H.; Khallaf, N.; Alonayzan, A. Building the Leeds Monolingual and Parallel Legal Corpora of Arabic and English Countries' Constitutions: Methods, Challenges and Solutions. *Corpus Pragmat.* **2023**, *7*, 103–119. [CrossRef]
12. Gupta, S.; Poplavska, E.; O'Toole, N.; Arora, S.; Norton, T.; Sadeh, N.; Wilson, S. Creation and analysis of an international corpus of privacy laws. *arXiv* **2022**, arXiv:2206.14169.
13. Jin, Y.; He, H. An artificial-intelligence-based semantic assist framework for judicial trials. *Asian J. Law Soc.* **2020**, *7*, 531–540. [CrossRef]
14. Qiu, M.; Zhang, Y.; Ma, T.; Wu, Q.; Jin, F. Convolutional-neural-network-based Multilabel Text Classification for Automatic Discrimination of Legal Documents. *Sens. Mater.* **2020**, *32*, 2659–2672. [CrossRef]
15. Sando, K.; Suzuki, T.; Aiba, A. A constraint solving web service for a handwritten Japanese historical kana reprint support system. In *Agents and Artificial Intelligence: 10th International Conference, ICAART 2018, Funchal, Madeira, Portugal, 16–18 January 2018*; Revised Selected Papers 10; Springer International Publishing: Berlin/Heidelberg, Germany, 2019.
16. Braz, F.A.; da Silva, N.C.; de Campos, T.E.; Chaves, F.B.S.; Ferreira, M.H.; Inazawa, P.H.; Coelho, V.H.; Sukiennik, B.P.; de Almeida, A.P.G.S.; Vidal, F.B.; et al. Document classification using a Bi-LSTM to unclog Brazil's supreme court. *arXiv* **2018**, arXiv:1811.11569.
17. Maken, P.; Gupta, A.; Gupta, M.K. A study on various techniques involved in gender prediction system: A comprehensive review. *Cybern. Inf. Technol.* **2019**, *19*, 51–73. [CrossRef]
18. Xiao, G.; Mo, J.; Chow, E.; Chen, H.; Guo, J.; Gong, Z. Multi-Task CNN for classification of Chinese legal questions. In Proceedings of the 2017 IEEE 14th International Conference on e-Business Engineering (ICEBE), Shanghai, China, 4–6 November 2017; IEEE: New York, NY, USA, 2017.
19. Alrasheed, N.; Prasanna, S.; Rowland, R.; Rao, P.; Grieco, V.; Wasserman, M. Evaluation of deep learning techniques for content extraction in Spanish colonial notary records. In Proceedings of the 3rd Workshop on Structuring and Understanding of Multimedia heritAge Contents, Chengdu, China, 20 October 2021.
20. Kirmizialtin, S.; Wrisley, D. Automated transcription of non-Latin script periodicals: A case study in the Ottoman Turkish print archive. *arXiv* **2020**, arXiv:2011.01139.
21. Noëmie, L.; Salah, C.; Vidal-Gorène, C. New Results for the Text Recognition of Arabic Maghrib {ī} Manuscripts—Managing an Under-resourced Script. *arXiv* **2022**, arXiv:2211.16147.
22. Bhusal, A.; Chhetri, G.B.; Bhattarai, K.; Pandey, M. *"Nepali OCR"*; Institute of Engineering, Pulchowk Campus: Lalitpur, Nepal, 2023. Available online: https://elibrary.tucl.edu.np/items/1c6b2978-08e9-4208-a521-371442a4d452 (accessed on 24 November 2024).
23. Shihab, I.H.; Hasan, R.; Emon, M.R.; Hossen, S.M.; Ansary, N.; Ahmed, I.; Rakib, F.R.; Dhruvo, S.E.; Dip, S.S.; Pavel, A.H.; et al. Badlad: A large multi-domain Bengali document layout analysis dataset. In *International Conference on Document Analysis and Recognition*; Springer Nature: Cham, Switzerland, 2023.
24. Alghamdi, M.A.; Alkhazi, I.S.; Teahan, W.J. Arabic OCR evaluation tool. In Proceedings of the 2016 7th International Conference on Computer Science and Information Technology (CSIT), Amman, Jordan, 13–14 July 2016; IEEE: New York, NY, USA, 2016.
25. Salehudin, M.; Basah, S.; Yazid, H.; Basaruddin, K.; Som, M.M.; Sidek, K. Analysis of optical character recognition using easyocr under image degradation. *J. Phys. Conf. Ser.* **2023**, *2641*, 012001. [CrossRef]

26. Rukundo, O. Effects of Image Size on Deep Learning. *Electronics* **2023**, *12*, 985. [CrossRef]
27. Menassel, Y.; Marie, R.R.; Abbas, F.; Gattal, A.; Al-Sarem, M. Dynamic Feature Weighting for Efficient Multi-Script Identification Using YafNet: A Deep CNN Approach. *J. Intell. Syst. Internet Things* **2025**, *14*, 260–273.
28. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; IEEE: New York, NY, USA, 2009.
29. Nugraha, G.S.; Darmawan, M.I.; Dwiyansaputra, R. Comparison of CNN's Architecture GoogleNet, AlexNet, VGG-16, Lenet-5, Resnet-50 in Arabic Handwriting Pattern Recognition. *Kinet. Game Technol. Inf. Syst. Comput. Netw. Comput. Electron. Control* **2023**, *8*, 545–554. [CrossRef]
30. Khudeyer, R.S.; Almoosawi, N.M. Combination of machine learning algorithms and Resnet50 for Arabic Handwritten Classification. *Informatica* **2023**, *46*, 9. [CrossRef]
31. Alyahya, H.M.; Ben Ismail, M.M.; Al-Salman, A. Intelligent ResNet-18 based Approach for Recognizing and Assessing Arabic Children's Handwriting. In Proceedings of the 2023 International Conference on Smart Computing and Application (ICSCA), Hail, Saudi Arabia, 5–6 February 2023; IEEE: New York, NY, USA, 2023.
32. Finjan, R.H.; Rasheed, A.S.; Hashim, A.A.; Murtdha, M. Arabic handwritten digits recognition based on convolutional neural networks with resnet-34 model. *Indones. J. Electr. Eng. Comput. Sci.* **2021**, *21*, 174–178. [CrossRef]
33. Bin Durayhim, A.; Al-Ajlan, A.; Al-Turaiki, I.; Altwaijry, N. Towards accurate children's Arabic handwriting recognition via deep learning. *Appl. Sci.* **2023**, *13*, 1692. [CrossRef]
34. Zendel, O.; Culpepper, J.S.; Scholer, F.; Thomas, P. Enhancing human annotation: Leveraging large language models and efficient batch processing. In Proceedings of the 2024 Conference on Human Information Interaction and Retrieval, Sheffield, UK, 10–14 March 2024.
35. Rubin, D.L.; Akdogan, M.U.; Altindag, C.; Alkim, E. ePAD: An image annotation and analysis platform for quantitative imaging. *Tomography* **2019**, *5*, 170. [CrossRef]