Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 03, September 2024.

A HYBRID MODEL FOR ARABIC SCRIPT RECOGNITION BASED ON CNN-CBAM AND BLSTM

Mohamed Dahbali, Noureddine Aboutabit and Nidal Lamghari

(Received: 4-Mar.-2024, Revised: 5-May-2024, Accepted: 21-May-2024)

ABSTRACT

Handwriting recognition, particularly for Arabic, is a very challenging field of research due to various complex factors, such as the presence of ligatures, cursive writing style, slant variations, diacritics, overlapping and other difficult problems. This paper specifically addresses the task of recognizing offline Arabic handwritten text lines. The main contributions include the pre-processing stage and the utilization of a deep learning-based approach with data-augmentation techniques. The pre-processing step involves correcting the skew of text-lines and removing any unnecessary white space in images. The deep-learning architecture consists of a Convolutional Neural Network and Convolutional Block Attention Module for feature extraction, along with Bidirectional Long Short-Term Memory for sequence modeling and Connectionist Temporal Classification as a decoder. Data-augmentation techniques are utilized on the images in the database to enhance the system's ability to recognize a wide range of Arabic characters and to extend the level of abstraction in patterns due to synthetic variations. Our suggested approach has the capability of precisely recognizing Arabic handwritten texts without the necessity of character segmentation, thereby resolving various issues associated with this aspect. The results obtained from the KHATT database highlight the effectiveness of our approach, demonstrating a Word Error Rate of 14.55% and a Character Error Rate of 3.25%.

KEYWORDS

Handwriting recognition, Arabic database, Data augmentation, CNN, BLSTM.

1. INTRODUCTION

The Arabic script is one of the most widely utilized scripts in the world. The Arabic script is a cursive script and is renowned for its complex and challenging behaviors in comparison to handwritten Latin script [1][2][3], as well as the presence of handwriting-style variations and position-dependent character shapes. Some difficulties of handwritten Arabic, which involves segmentation into optical units, such as words, sub-words and characters, the overlapping of characters, the inclination of text lines, historical documents and the lack of publicly accessible databases, make the development of a method for recognizing handwritten Arabic text a serious challenge. It will be advantageous to design systems capable of resolving these issues and profiting from their applications, such as postal code/zip recognition, form processing, automatic cheque processing in banks …etc. [4].

Hidden Markov Model (HMM) was initially used for speech recognition due to its ability to model sequences, but in recent years, it has been studied for use in handwriting-recognition systems. There are numerous reasons for the success of HMM in text recognition, including the fact that it provides the advantage of joint segmentation and recognition [5] and has mathematical and theoretical bases. In recent years, researchers have focused on modeling Arabic handwritten sentences using techniques of deep learning. Several deep-learning models, such as Recurrent Neural Network (RNN), Long Short-term Memory (LSTM), Bi-directional Long Short-term Memory (BLSTM) and Multi-dimensional Long Short-term Memory (MDLSTM), can be utilized to model a sequence of data. Among the shortcomings of deep-learning models is their reliance on a large quantity of image data to converge and generalize, as well as their tendency to overfit training data. This problem could be avoided by incorporating data augmentation into the training process. This method can enhance the presentation of optical units, such as characters, sub-words and words, in handwritten Arabic text-line images. The primary advantage of modeling text at the character level is that the system can recognize out-of-vocabulary optical units in the test set and generalize the recognition system for unseen data. The method adopted during feature extraction is crucial for any recognition system to effectively model sentences. Two main approaches exist for extracting features: hand-crafted features, such as HOG [6] and LBP [7] descriptors and learned

M. Dahbali, N. Aboutabit and N. Lamghari are with LIPIM laboratory, National School of Applied Sciences, Sultan Moulay Slimane University, Beni Mellal, Morocco. Emails: dahbali.mohamed@gmail.com, n.aboutabit@usms.ma and n.lamghari@usms.ma

features, such as those extracted by CNN [8]. This research is driven by the need to examine the impact of refining features extracted from CNN. Convolutional Block Attention Module (CBAM) is used to achieve this objective. The module sequentially infers attention maps along two different dimensions, channel and spatial and then multiplies them with the input feature map for adaptive-feature refinement. Our study aims to explore how refining the features extracted from CNN using Convolutional Block Attention Module (CBAM) can improve the recognition of Arabic handwritten text lines. In addition, our goal is to evaluate the impact of incorporating data-augmentation techniques. It is worth mentioning that, as far as we know, no prior research in the field of handwriting text-recognition systems has suggested the integration of CBAM into their systems. This highlights the originality and importance of our approach.

The primary contributions of this work include the proposal of a new technique for the pre-processing stage and expanding the size of the database through the use of data-augmentation techniques. Additionally, the utilization of CBAM as an efficient attention module for feedforward convolutional neural networks. This module not only reduces the number of parameters and computational power required, but also enhances the performance of the proposed system. Furthermore, the extracted features are used as input for a BLSTM and the output is then passed to the CTC layer. Finally, the evaluation of the proposed system is conducted using Character Error Rate (CER) and Word Error Rate (WER).

The paper is organized as follows: we review related works in Section 2 and introduce our proposed system in Section 3. Section 4 provides a comprehensive description of the database utilized in this study, as well as the data-augmentation techniques, evaluation methods and implementation details. Section 5 presents the findings and analysis, along with a comparative evaluation of our approach with other existing techniques. The conclusion of the paper is presented in Section 6.

2. RELATED WORKS

The literature contains works on Arabic handwriting-recognition systems employing HMMs and Deep Learning methods. The authors of [9] presented a comprehensive Arabic offline handwritten-text database (KHATT) encompassing all Arabic character forms. They proposed an HMM-based recognition system. Image adaptive pixel density and horizontal and vertical edge derivatives using the sliding window technique were utilized during the phase of feature extraction. The recognition is accomplished using HMM and HTK tools. The work [10] evaluated the quality of three types of language models (LMs) based on full word, hybrid word/Part-of-Arabic-Word (PAW) and full PAW by utilizing the Maurdor and Khatt databases. For the recognition system, they utilized hybrid HMM/Multi-directional LSTM Recurrent Neural Networks (MDLSTM-RNNs). The authors of [11] employs the KHATT database and proposes a new architecture based on MDLSTM, which consists primarily of three MDLSTM hidden layers and tanh layers. Another work [12] discussing the application of a Multi-directional LSTM Recurrent Neural Network (MDLSTM-RNN) based deep-learning model. The authors proposed examining the effect of utilizing various optical units with a hybrid word/part-of-Arabic word language model by training the optical model on Fixed and Random paragraphs from the KHATT database. In [13], another attempt to recognize handwritten Arabic text lines was made using MDLSTM. In this work, the authors proposed to capitalize on the visual similarities between Arabic, Urdu and Pashto. For this purpose, they examined a variety of combinations of these languages utilizing diverse optical models. Due to the lack of available datasets with real data for the Urdu language, the suggestion made in this work is to generate synthetic images for the three languages, so that meaningful comparisons can be made between the results of the optical models. For Arabic language in this experiment, the KHATT database used. The authors of the paper [14] proposed a novel technique for recognizing handwritten Arabic text line images using the KHATT database. For the feature extraction phase, segment-based and distribution-concavity (DC) based features were used and a 3-gram language model was employed for post-processing. Low-level, mid-level and high-level combinations of the BLSTM network were proposed. Experiments were conducted on two scenarios: the first employs a lexicon consisting of all tokenized words extracted from the KHATT corpus and the second employs a lexicon limited to words occurring in the training corpus. The authors of [15] utilized a multi-stage HMM-based text-recognition system for handwritten Arabic. They separated the core shapes of characters from their diacritics first. These Arabic core shapes are then converted into sub-core shapes to reduce the number of required models for modeling

Arabic characters. The models for multi-stage recognition system are sub-core shapes and diacritics. The proposed system was evaluated using the KHATT and IFN/ENIT databases. Another attempt for the recognition of handwritten Arabic text line images from the KHATT database is examined in [16]. This work employs a MDLSTM as an optical model. The main purpose of this work is to investigate the effect of data augmentation applied to each instance of text line image on the training of this optical model. Cross-validation based on a statistical process is used to evaluate the performance of the proposed system. The authors of [17] proposed a new architecture that combines CNN and BLSTM. Experiments are conducted on full text line images from the KHATT database. In [18], a unified end-to-end model for paragraph text recognition using hybrid attention is proposed. This module can be divided into three parts: an encoder that extracts feature maps from the entire paragraph image. Next, an attention module iteratively produces a vertical weighted mask that allows for focusing on the features of the current text line. A decoder module then recognizes the character sequence, resulting in the recognition of an entire paragraph. The suggested approach, applied to three widely-used datasets (RIMES, IAM and READ2016), produced state-of-the-art character error rates at the paragraph. The authors of [19] emphasized the significance of the encoder representation in enhancing the efficiency of Handwritten Text Recognition (HTR) systems. The authors suggested an encoder-decoder architecture for HTR that merges the advantages of local and global cross-channel attention to enhance the representation of the encoder. The experimental results on the IAM dataset demonstrate a significant decrease in CER and WER when the proposed module is implemented in the state-of-the-art HTR Flor model and Puigcerver model, respectively. The authors of [20] presented an encoder-decoder model for recognizing offline handwritten text. The DenseNet encoder is used to extract multiscale features. A contextual attention localizer was implemented between two gated recurrent units in order to integrate the role of context in the reading process and focus on particular characters. The model was assessed using the KHATT database and demonstrated superior performance compared to both simple-attention and multi-directional LSTM models. In [21], a new method for offline Arabic handwritten-text recognition is presented. The proposed system combines a CNN and a BLSTM followed by a CTC. Additionally, the authors introduce an algorithm for data augmentation to enhance the quality of the data. Significant performance was attained when compared to other models on the KHATT database. The authors of [22] examined two different end-to-end architectures for the recognition of Arabic handwritten-text lines: The transformer transducer and the standard transformer architecture with cross-attention. They generated a synthetic dataset consisting of printed Arabic text-line images, along with their corresponding ground truth, by utilizing different open-source fonts. The models conducted training using the synthetic dataset and were subsequently fine-tuned using the original dataset in order to assess their performance. The researchers discovered that both models exhibit strong competitiveness, with the cross-attention transformer achieving superior accuracy and the transformer transducer demonstrating faster processing speed when using the KHATT database.

As previously stated, numerous approaches have been developed to enhance offline Arabic handwritten-text recognition systems by increasing the depth or width of neural networks. While these approaches have achieved favourable results, the neural networks are excessively deep or wide, posing a significant computational challenge for computers. Motivated by the shortcomings of previous studies, we introduce a new and efficient feature representation of a lightweight CNN using CBAM. The objective of utilizing CBAM is to improve the accuracy of the proposed system while simultaneously minimizing computational requirements.

3. PROPOSED SYSTEM

This section provides an overview of the proposed system, which primarily comprises three different stages, as presented in Figure 1. The initial stage encompasses the pre-processing of the input image. Subsequently, features are extracted from the images utilizing CNN and CBAM. As the final stage, sequence modeling is done utilizing BLSTM and CTC as decoder. The components of the proposed system are detailed below.

3.1 Pre-processing

Pre-processing is a critical stage in addressing the issues associated with the text-lines of KHATT database, particularly the correction of text line slant and the elimination of extra white space in images. To correct the slant of text lines, we have utilized a technique based on horizonal projection inspired by

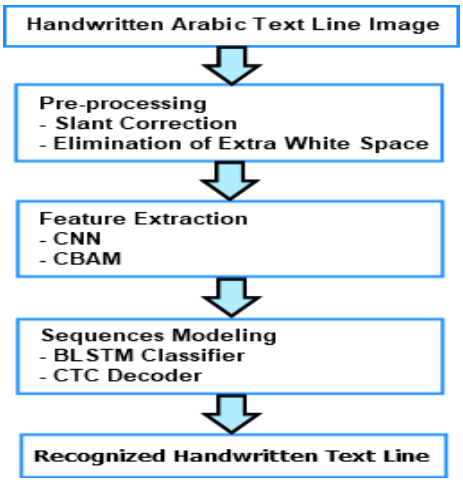
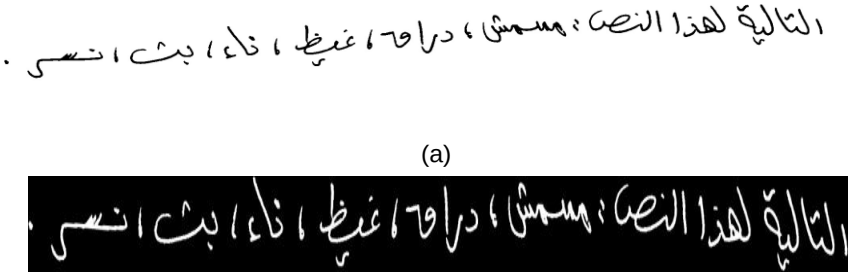


Figure 1. Proposed-system steps.

[23]. Diverse angle points are projected into an accumulator array, where the skew angle is defined as the projection angle within a search interval that maximizes alignment. To extract this angle, we first selected a testing range of angles, then rotated images using the selected angles and generated histograms and then computed a score based on the difference between peaks, with the angle with the highest score being the skew angle. During our experiments, we used a rotation angle ranging between -5 and 5. Pixels that do not belong to the text contribute to the existence of extra space, which degrades the performance of the system overall. We followed three steps based on morphological operations to resolve this issue. First, we dilated the handwritten text in images using a rectangular structuring element both vertically and horizontally and then we sorted all of the shapes in the images by contour space with their locations. Finally, we kept the shape with the most space and extracted text lines based on its location. Figure 2 depicts an example of a pre-processed text-line image.



(a)

(b)

Figure 2. (**a**) Original sample. (**b**) Image after pre-processing.

3.2 Feature Extraction Technique

Feature extraction is the most important element of any system designed to recognize handwritten text. CNN and CBAM were utilized in this study. CNN is known for its capacity to extract features from images using convolution layers. Typically, CNN consists of a series of convolution layers followed by down-sampling layers, such as max pooling. Depending on the size of the filter, the convolution layers divide the image in entry into partial images and extract features. Based on the number of filters utilized, a feature map is generated after each convolution operation. As the number of convolution layers increases, more robust features are obtained, which improves the performance of the recognition system. We used Rectified Linear Unit (RELU) activation function. There are three main advantages to using this function: the computational simplicity, because it is based on a max function, the representational sparsity to accelerate and simplify network learning, because it can output a true zero value, unlike other activation functions such as tanh and sigmoid and the linear behaviour, which aids in network optimization and prevents the vanishing of gradient.

CBAM [24] is a method for enhancing a recognition system by applying adaptive refining of CNN-extracted feature map. This method can be divided into two modules, specifically Channel Attention Module (CAM) and Spatial Attention Module (SAM). In this technique, CAM is implemented before

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 03, September 2024.

SAM. CAM is an architecture with minor differences to Squeeze Excitation Module (SEM) proposed in [25]. It is ideal to illustrate the procedure of SEM in order to comprehend the significance of CAM. All SEM operations can be broken down into three steps in order to extract channel weights. Initially, every channel is reduced to a single pixel. In the second step, a multi-layer perceptron (MLP) with a bottleneck is employed, followed by a sigmoid activation layer in the third step. To accomplish the first step, Global Average Pooling (GAP) operation is performed on the channels to aggregate spatial information into a single pixel. The outcome of the first step is a 1-D vector that will be utilized in the subsequent step. The second step is to use the vector output from the first step as input to a MLP network in which the bottleneck size is constrained by the reduction ratio R. The ratio of the number of channels N to the reduction ratio R is used to calculate the total number of neurons in a bottleneck. The greater the reduction ratio, the smaller the bottleneck. At the third step, the output vector from this MLP is passed to a sigmoid activation layer to maintain channel-weight values between 0 and 1.

The main distinction between SEM and CAM is the addition of Global Max Pooling (GMP). The output of the convolution operation is a feature map with the dimensions C×H×W, where H represents the height of each channel, W represents the width of each channel and C represents the total number of channels. Using GAP and GMP, the feature map generated by CNN is transformed into two consecutive vectors of size C×1×1. GMP preserves the contextual information in the form of edges presented in images. The combination of the two pooling operations provides more information than using GAP alone in SEM. The two vectors are successively passed to MLP. It is important to note that the same weights are used for both vectors in MLP. The final vector is constructed by applying an element-wise sum of the two MLP outputs and is then passed to the sigmoid layer to generate channel weights, which is used to distribute weights between channels in the feature map using an element-wise product. Figure 3 depicts the working process of CAM.

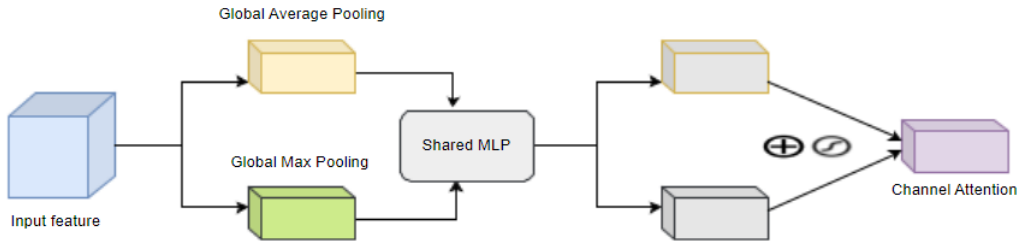


Figure 3. Diagram of channel-attention module (CAM).

The main aim of SAM is to generate an attention mask and apply it to the feature map to obtain more accurate features. The construction of attention mask consists of three sequential steps. The initial operation is a down-sampling operation. The objective of this step is to reduce the dimensions of feature map from C×H×W to 2×H×W by applying average-pooling and max-pooling operations along the channel axis and concatenate them. The resulting feature descriptor is then forwarded to the next step, which consists of a convolution-layer operation with a filter of size 7 × 7 and employs padding to do the same. In the third step, the output is sent to a sigmoid activation layer to map the mask values to the range from 0 to 1. To improve features, an element-wise product is performed between the current feature map and the resulting one channel output. Figure 4 depicts the working process of SAM.

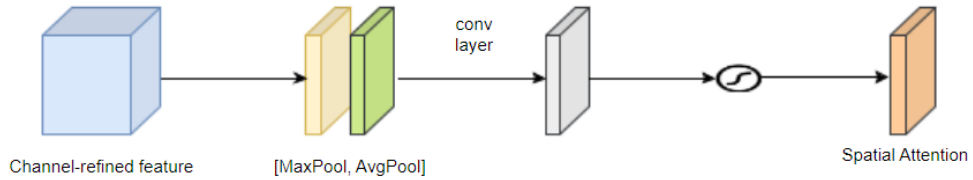


Figure 4. Diagram of spatial-attention module (SAM).

3.3 Optical Model

Diverse handwriting-recognition systems based on deep learning can be used for the recognition of handwritten Arabic text lines. BLSTM served as the optical model for our study. Because RNN is particularly susceptible to exploding/vanishing gradients, the LSTM model and its variants were

developed to address these issues. The addition of a new layer to maintain long-distance connections between optical units presented in sequence is the primary distinction between standard LSTM and BLSTM. In BLSTM, the input flow is bidirectional, which allows the system to model the relationship between sequence elements from the past to the future and *vice versa*. BLSTM consists of four main components: a forget gate, an input gate, an output gate and a cell state. These gates play a crucial role in the network, acting as filters to determine which useful data should be transmitted to subsequent steps. As its name suggests, the forget gate uses the previous hidden state and new input to determine what is relevant to keep from the prior cell state. Using the same parameters as the forget gate, the input gate identifies the information that should be updated in the current cell state. The output gate specifies the current hidden state that will be transmitted to the following LSTM unit. The output of BLSTM is input to a Softmax layer that contains a number of units equal to the size of characters in the vocabulary, plus a special character called blank to solve the problem of duplicate characters when encoding text and to create different alignments for the same text. The activation of L units is interpreted as the probability of observing the corresponding characters per time step and can be represented as a matrix. These output units specify the probabilities of all possible character-sequence alignments with the input sequence. The CTC layer comes at this point to compute the loss value for training the network. The advantage of CTC is that it avoids the explicit segmentation of Arabic text, which is notoriously difficult, particularly at the character level. Decoding is the process of determining the optimal alignment possible given the distribution over the output. Numerous approaches, such as best-path search and beam search, are proposed for approximating the most precise optimal alignment. Given the output probability matrix, the best-path search approach considers the character with the highest probability at each time step. The primary advantage of this strategy is the speed of decoding, while the disadvantage is a significant chance of failing to predict the correct ground truth. The beam-search method is based on a single hyper-parameter known as beam width, which specifies the number of characters with the highest predicted probabilities selected at each time step. Each predicted character in the first-time step will be the first character of output sequences, respectively. At each subsequent time step, based on the candidate output sequences from the previous time step, we continue to select candidate output sequences with the highest predicted probabilities until the final time step. The main advantage of this strategy is expanding space search and decoding the output close enough to the optimal sequence, while the primary drawback is the increased time required for decoding in comparison to greedy search. To extract the character sequence after decoding, all blanks and repeated characters are removed from the predictions.

4. EXPERIMENTAL SETUP

In this section, we initially present the database employed for this study. Next, we describe the methods utilized to increase the size of the database by adding extra synthetic images during the data-augmentation stage. Next, we outline the evaluation methods, followed by an explanation of the implementation details.

4.1 Database

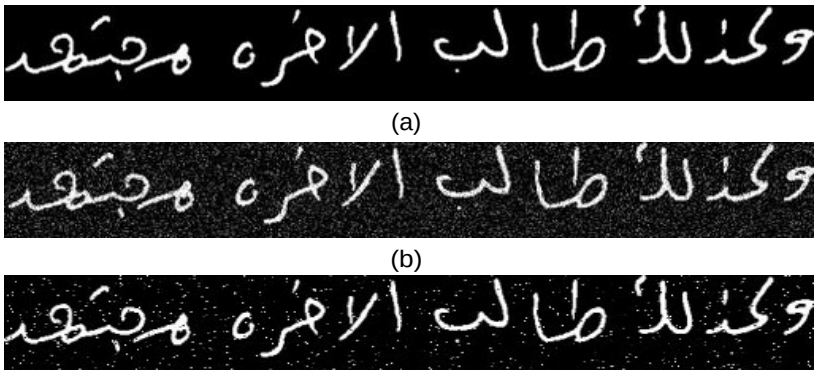
KHATT is an Arabic offline handwritten-text database that is freely accessible for academic research. This database was created to address the lack of Arabic handwritten-text datasets. The database has 1000 forms written by different writers and each form has 4 pages [9]. It can be used for research in a variety of fields, including text recognition, writer identification and verification, form analysis, segmentation, …etc. One thousand writers from various countries were engaged to fill out four-page forms. The first page contains writer information, while the second page contains both fixed and randomly-selected paragraphs. Fixed paragraphs cover all Arabic character shapes and randomly-selected paragraphs were collected from a large corpus developed from 46 sources to create a database that is a statistical representation of the corpus, with each paragraph being unique across all forms. The third page includes another unique randomly-selected paragraph and the same fixed paragraph as the second page. The fourth page contains free writing on a variety of topics with ruled lines. For experimental purposes, we construct two partitions from the database. The first partition consists of a merged dataset that includes both unique and fixed-text lines. The second partition contains only unique text lines. Table 1 presents statistical information regarding the database partitions utilized in the evaluation of the proposed system.

Table 1. Statistical information pertaining to the data utilized in this study.

Partition	**Number of text lines**	**Number of text lines after data augmentation**
Unique text lines + Fixed text lines	Train: 9.470 Test: 2.001	Train: 28.410 Test: 6.003
Unique text lines	Train: 4825 Test: 960	Train: 14.475 Test: 2.880

4.2 Data Augmentation

The variability of data plays a significant role in training models for classification and sequence-modeling tasks, as well as improving the generalization capability of systems, particularly in the case of deep-learning models. To improve the performance of the proposed system, we used data augmentation to expand the size of the database by creating new image samples. Additive Gaussian Noise (AGN) and Salt and Pepper Noise (SPN) were utilized to add noise to the images. In the case of AGN, we used a normal distribution with a mean of 0 and a variance of 50 and for SPN, we covered 10% of all pixels in the image. Figure 5 illustrates an example of the applied augmentation method.



(a) (b) (c)

Figure 5. (**a)** Original sample. (**b**) AGN applied on image and (**c)** SPN applied on image.

4.3 Evaluation Methods

4.3.1 Cross-validation Method

The effectiveness of a deep-learning model is determined by its ability to generalize. It represents the ability of the system to recognize unseen images containing handwritten Arabic text lines. For this purpose, we utilized a statistical technique known as k-fold cross-validation to evaluate the performance of our proposed system. The proposed method divides the entire dataset into folds according to a parameter k that specifies the number of folds, using one fold for testing and the remaining folds for training. Each sample is eligible for use in training and evaluating the model. The primary benefit of using such a method is avoiding biased and overly optimistic results due to the nature of the utilized data. The most difficult aspect of this method is determining an adequate value for the parameter k to create a training and test-set partition that is statistically representative of the entire dataset. The performance metric derived from k-fold cross validation is summarized by the mean model skill scores.

4.3.2 Optical Model-evaluation Method

The evaluation of optical model performance requires the use of reliable evaluation metrics. CER and WER are two metrics that have been utilized previously in the literature. CER and WER represent the error of predicted labels at the character and word levels, respectively. These metrics are based on the Levenshtein distance [26] concept. In the case of CER, the error between the prediction and the ground truth is determined by calculating the ratio of insertions I, substitutions S and deletions D relative to the total number of characters NC in the ground truth, as shown in the following formula:

$$CER(\%) = \frac{S+I+D}{NC} \times 100 \quad (1)$$

In the case of WER, the error between the prediction and the ground truth is determined by calculating the ratio of insertions I, substitutions S and deletions D relative to the total number of words NW in the ground truth, as shown in the following formula:

$$WER(\%) = \frac{S+I+D}{NW} \times 100 \quad (2)$$

4.4 Implementation Details

The input to our system consists of grayscale images. All images are resized to dimensions of 64×1024, with 64 representing the height and 1024 representing the width. In the CNN architecture, we utilized 5 convolution layers with a filter size of 3×3. Additionally, we added 4 down-sampling layers with the max-pooling operation. The purpose of the first and second pooling operations is to reduce the vertical and horizontal size of the feature map. The last two pooling operations only reduce vertical size of the feature map. Figure 6 depicts the employed architecture. At the end of CNN architecture, we have 128 channels with a size of 4×256 that will serve as input to CBAM. The reduction ratio for CAM was defined as 2. The filter size employed by SAM is 7, identical to the original paper [24]. The features generated by the CBAM architecture are fed to the BLSTM. To model handwritten Arabic text lines, we utilized two distinct BLSTMs, each of which consisted of 256 LSTM units in both directions (i.e., 128 LSTM units in either direction). The output was decoded using the beam-search algorithm. We evaluated a number of beam-width values before settling on 10, because it offers the best performance. The Adam optimizer is employed for training with a batch size of 50. To assess the proposed system, we employed the k-fold cross-validation technique with a k-value of 5.

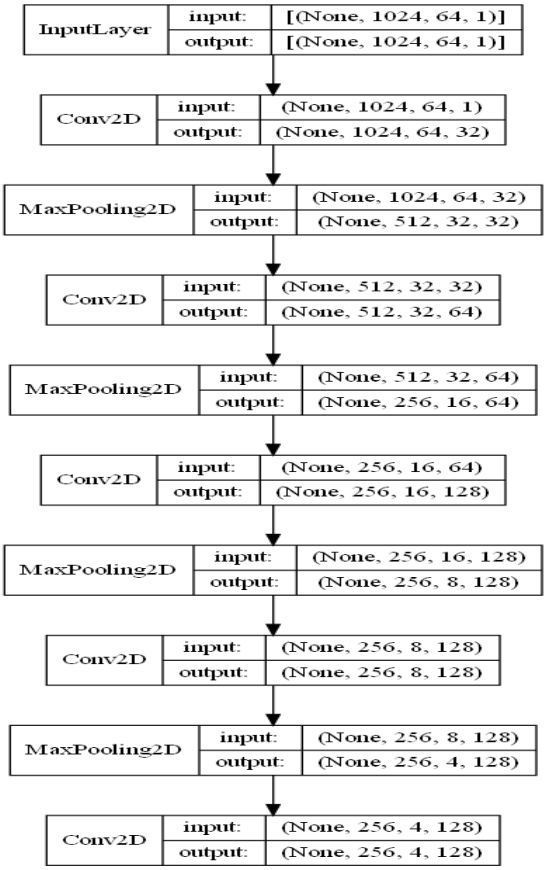


Figure 6. CNN architecture for feature-map extraction.

5. Results and Discussion

To assess the efficacy of the proposed system, we utilized two networks, one without CBAM and one with CBAM, employing two different approaches from KHATT database, with and without data augmentation. We performed two experiments during this study. The first experiment involved the merged dataset, which consisted of both fixed and unique text lines. The second experiment focused solely on the unique text lines.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 03, September 2024.

Table 2 demonstrates that in the first experiment, the CER decreased from 13.26% to 13.00% and the WER decreased from 45.33% to 44.49% as a result of incorporating CBAM into the proposed system. The use of data augmentation has resulted in a major enhancement of the proposed system, leading to a significant reduction in both CER and WER.

Table 2. Performance of the system in the first experiment using CER and WER.

	CER	**WER**
CNN	13.26%	45.33%
CNN+CBAM	13.00%	44.49%
CNN+CBAM+Data Augmentation	3.25%	14.55%

Table 3 demonstrates that, similar to the initial experiment, the CER decreased from 19.84% to 19.02% and the WER decreased from 67.78% to 64.67% when the CBAM was incorporated into the proposed system during the second experiment. Data augmentation was utilized in this experiment, leading to a significant improvement of the proposed system, akin to the initial experiment.

Table 3. Performance of the system in the second experiment using CER and WER.

	CER	**WER**
CNN	19.84%	67.78%
CNN+CBAM	19.02%	64.67%
CNN+CBAM+Data Augmentation	3.96%	18.57%

The primary distinction between fixed and unique text-line images is text content. Fixed text-line images consist of similar texts encompassing all Arabic character shapes, whereas unique text line images are made up of different texts. The presence of similar texts containing similar optical units in the dataset may allow the system to model sequences efficiently. However, the presence of distinct texts with distinct optical units in the dataset renders the system incapable of effectively modeling long-term dependencies. The distance between relevant information in the entire sequence and current time step information may be too great, thereby degrading the performance of the system, as in the case of unique text line images containing distinct texts. It is necessary to adopt a strategy to resolve this issue. We suggested employing data augmentation to increase the presence of optical units in the dataset and to enhance the level of abstraction in patterns due to synthetic variations. CBAM ensures that CNN focuses on useful features presented on images and avoids non-useful background information. This is achieved by exploiting the inter-channel and inter-spatial relationships of features through the integration of channel and spatial-attention modules, respectively. BLSTM was chosen as the optical model for our method. The primary distinction between the conventional LSTM and the bidirectional LSTM lies in the inclusion of an extra layer that handles the opposite direction of the sequence. This enables the modeling of sequential dependencies among characters and words in both the forward and backward directions of the sequence. CER and WER values for the unique text lines are higher than those for the merged dataset, demonstrating that similar text in the dataset improves the performance of the optical model. The inclusion of CBAM in the network resulted in a notable enhancement in the performance of the proposed system, leading to a reduction in both CER and WER by a significant margin when compared to the network without CBAM. Undoubtedly, the CBAM-CNN architecture produces superior quality features compared to those generated solely by CNN. To confirm this hypothesis, we can verify it by comparing the CER and WER results of our work with the results of recent studies [17][21] that employed a CNN-BLSTM architecture. Table 4 depicts a comparative study between our work and other works that utilized the same database. It is apparent that the proposed CNN-CBAM-BLSTM system is successful, as it outperforms other recent works that utilized the same database.

Table 4. Comparison with other state-of-the-art systems evaluated on the KHATT database.

Authors	**Features**	**Optical model**	**No. of text lines**	**Character Error Rate (CER)**	**Word Error Rate (WER)**
Mahmoud et al. [9]	Image adaptive pixel density features and horizontal and vertical edge derivatives	HMM using HTK tools	Train: 4808 Test: 966 Validation: 938	53.87%	
Ben Zeghiba et al. [10]	Raw pixels	HMM/MDLSTM-RNN	Train: 4428 Test: 959 Validation: 876		30.9%
Ahmed et al. [11]	Raw pixels	MDLSTM	Train: 4825 Test: 966 Validation: 937	24.25%	
Ben Zeghiba [12]	Raw pixels	MDLSTM-RNN	-Random paragraphs -Fixed paragraphs -Fixed and random paragraphs		41.4% 8.3% 23.0%
Ahmad et al. [13]	Raw pixels	MDLSTM	Train: 4825 Test: 966 Validation: 937	24.30%	
Jemni et al. [14]	Segment-based and distribution-concavity (DC)-based features	Combination of BLSTM	Train: 9475 Test: 2007 Validation: 1901	7.85%	13.52%
Ahmad and Fink [15]	Feature-based on ink pixels	Multi-stage HMM system	Train: 4808 Test: 966 Validation: 937	41.21%	
Ahmad et al. [16]	Raw pixels	MDLSTM	Train: 4825 Test: 966	19.98%	
Anjum and Khan [20]	DenseNet encoder	GRUs with Contextual Attention Localization	Train: 4825 Test: 966 Validation: 937	22.85%	64.17%
Momeni and Baba Ali [22]	Transformer encoder	Transformer with Cross-attention and Transformer Transducer	6742 of unique text lines	18.45%	
Noubigh et al. [17]	CNN	BLSTM	Train: 8505 Test: 1867 Validation: 1584	15.8%	35.6%
Lamtougui et al. [21]	CNN	BLSTM	Train: 4825 Test: 966		19.85%
Proposed system	CNN + CBAM	BLSTM	Train: 4825 Test: 960	3.96%	18.57%
			Train: 9.470 Test: 2.001	3.25%	14.55%

6. CONCLUSION

In this paper, we addressed the recognition of handwritten Arabic text lines, one of the most challenging problems associated with Arabic optical models. We used the database KHATT, which is renowned for the complexity of its handwriting, along with the data-augmentation technique to generate more synthetic images. Our study is the first to use CBAM to evaluate the effect of attention modules on CNN-generated feature map. Deep-learning architectures are the state-of-the-art systems used to model sequences in the literature over the past few years. BLSTM architecture was used to model Arabic

handwriting in our study. To evaluate our CNN-CBAM-BLSTM architecture, we conducted two experiments, the first with a dataset containing both unique and fixed text-line images and the second with unique text-line images. In the first experiment, the CER was 3.25% and the WER was 14.55%. In the second experiment, the CER was 3.96% and the WER was 18.57%. The reported results are comparable to those of recent studies that utilized the same database. Although our model performed well, the challenge of decreasing the WER presents an opportunity for future investigation. Furthermore, it will be essential to enlarge the database to include a wider range of handwritten Arabic text-line images in order to improve the model's generalizability and reduce any possible biases.

REFERENCES

[1] S. Ahmed, S. Naz, S. Swati, M. I. Razzak and A. I. Umar, "UCOM Offline Dataset: An Urdu Handwritten Dataset Generation," Int. Arab Journal of Information Technology, vol. 14, no. 2, pp. 239-245, 2017.

[2] A. Graves and J. Schmidhuber, "Offline Handwriting Recognition with Multidimensional Recurrent Neural NetWorks," Proc. of the 21st Int. Conf. on Neural Information Processing Systems (NIPS 2008), Red Hook, pp. 545-552, 2009.

[3] S. Naz et al., "The Optical Character Recognition of Urdu-like Cursive Scripts," Pattern Recognition, vol. 47, no. 3, pp. 1229–1248, DOI: 10.1016/j.patcog.2013.09.037, Mar. 2014.

[4] S. Faisal Rashid, M.-P. Schambach, J. Rottland and S. von der Null, "Low Resolution Arabic Recognition with Multidimensional Recurrent Neural Networks," Proc. of the 4th Int. Workshop on Multilingual OCR (MOCR '13), Article no. 6, pp. 1-5, DOI: 10.1145/2505377.2505385, Aug. 2013.

[5] D. Xiang, H. Yan, X. Chen and Y. Cheng, "Offline Arabic Handwriting Recognition System Based on HMM," Proc. of the 2010 3rd IEEE Int. Conf. on Computer Science and Information Technology, DOI: 10.1109/iccsit.2010.5564429, Chengdu, Jul. 2010.

[6] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," Proc. of the 2005 IEEE Computer Society Conf. on Computer Vision and Pattern Recogn. (CVPR'05), vol. 1, pp. 886–893, 2005.

[7] T. Ojala, M. Pietikäinen and D. Harwood, "A Comparative Study of Texture Measures with Classification Based on Featured Distributions," Pattern Recognition, vol. 29, no. 1, pp. 51–59, Jan. 1996.

[8] Y. LeCun and Y. Bengio, "Convolutional Networks for Images, Speech and Time Series," Part of Book: The Handbook of Brain Theory and Neural Networks, p. 3361, 1995.

[9] S. A. Mahmoud et al., "KHATT: An Open Arabic Offline Handwritten Text Database," Pattern Recognition, vol. 47, no. 3, pp. 1096–1112, DOI: 10.1016/j.patcog.2013.08.009, Mar. 2014.

[10] M. F. Ben Zeghiba, J. Louradour and C. Kermorvant, "Hybrid Word/Part-of- Arabic-Word Language Models for Arabic Text Document Recognition," Proc. of the 2015 13th IEEE Int. Conf. on Document Analysis and Recognition (ICDAR), pp. 671-675, DOI: 10.1109/icdar.2015.7333846, Aug. 2015.

[11] R. Ahmad, S. Naz, M. Zeshan Afzal, S. Faisal Rashid, M. Liwicki and Dengel, "KHATT: A Deep Learning Benchmark on Arabic Script," Proc. of the 2017 14th IEEE IAPR Int. Conf. on Document Analysis and Recognition, pp. 10-14, DOI: 10.1109/icdar.2017.358, Nov. 2017.

[12] M. F. BenZeghiba, "A Comparative Study on Optical Modeling Units for Off-line Arabic Text Recognition," Proc. of the 2017 14th IEEE IAPR Int. Conf. on Document Analysis and Recognition (ICDAR), DOI: 10.1109/icdar.2017.170, Nov. 2017.

[13] R. Ahmad, S. Naz, M. Zeshan Afzal, S. Faisal Rashid, M. Liwicki and A. Dengel, "The Impact of Visual Similarities of Arabic-like Scripts Regarding Learning in an OCR System," Proc. of the 2017 14th IEEE IAPR Int. Conf. on Document Analysis and Recognition (ICDAR), DOI: 10.1109/icdar.2017.359, 2017.

[14] S. Khamekhem Jemni, Y. Kessentini, S. Kanoun and J.-M. Ogier, "Offline Arabic Handwriting Recognition Using BLSTMs Combination," Proc. of the 2018 13th IAPR Int. Workshop on Document Analysis Systems (DAS), DOI: 10.1109/das.2018.54, Apr. 2018.

[15] I. Ahmad and G. A. Fink, "Handwritten Arabic Text Recognition Using Multi-stage Sub-core-shape HMMs," Int. Journal on Document Analysis and Recognition (IJDAR), vol. 22, no. 3, pp. 329–349, 2019.

[16] R. Ahmad, S. Naz, M. Afzal, S. Rashid, M. Liwicki and A. Dengel, "A Deep Learning based Arabic Script Recognition System: Benchmark on KHAT," Int. Arab Journal of Information Technology, vol. 17, no. 3, pp. 299–305, DOI: 10.34028/iajit/17/3/3, May 2020.

[17] Z. Noubigh, A. Mezghani and M. Kherallah, "Contribution on Arabic Handwriting Recognition Using Deep Neural Network," Proc. of the Int. Conf. on Hybrid Intelligent Systems, Part of the Book Series: Advances in Intelligent Systems and Computing, vol. 1179, pp. 123–133, DOI: 10.1007/978-3-030-49336-3_13, 2020.

[18] D. Coquenet, C. Chatelain and T. Paquet, "End-to-End Handwritten Paragraph Text Recognition Using a Vertical Attention Network," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 1, pp. 508–524, DOI: 10.1109/tpami.2022.3144899, Jan. 2023.

[19] B. N. Shashank, S. Nagesh Bhattu and Krishna, "Improvising the CNN Feature Maps through Integration of Channel Attention for Handwritten Text Recognition," Communications in Computer and Information

Science, pp. 490–502, DOI: 10.1007/978-3-031-31417-9_37, Jan. 2023.
[20] T. Anjum and N. Khan, "CALText: Contextual Attention Localization for Offline Handwritten Text," arXiv.org, arXiv: 2111.03952, DOI: 10.48550/arXiv.2111.03952, 2021.
[21] H. Lamtougui, H. El Moubtahij, H. Fouadi and K. Satori, "An Efficient Hybrid Model for Arabic Text Recognition," Computers, Materials & Continua, vol. 74, no. 2, pp. 2871–2888, 2023.
[22] S. Momeni and B. Baba Ali, "A Transformer-based Approach for Arabic Offline Handwritten Text Recognition," Signal, Image and Video Processing, vol. 18, no. 4, pp. 3053–3062, Jan. 2024.
[23] J. Kanai and A. Bagdanov, "Projection Profile Based Skew Estimation Algorithm for JBIG Compressed Images," Int. J. on Document Analysis and Recognition, vol. 1, pp. 43–51, 1998.
[24] S. Woo, J. Park, J.-Y. Lee and I. S. Kweon, "CBAM: Convolutional Block Attention Module," arXiv.org, arXiv: 1807.06521, DOI: 10.48550/arXiv.1807.06521, Jul. 2018.
[25] J. Hu, L. Shen, S. Albanie, G. Sun and E. Wu, "Squeeze-and-Excitation Networks," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 42, no. 8, pp. 1–1, 2019.
[26] V. I. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions and Reversals," Soviet Physics-Doklady, vol. 10, no. 8, pp. 707-710, Feb. 1966.

ملخص البحث:

إنّ تمييز الكتابة بخطّ اليد، وبخاصّة باللّغة العربية، يُعدّ مجالاً بحثياً ينطوي على الكثير من التّحدّيات. ويرجع ذلك إلى عددٍ من العوامل المعقّدة، منها وجود علامات الشَّكْل، ونمط الكتابة، واختلاف اللّهجات، والتّداخل، إلى جانب العديد من المشكلات الصّعبة الّتي تكتنف تمييز الكتابة اليدوية.

تتناول هذه الورقة بالتّحديد أسطُراً مكتوبة بخطّ اليد باللّغة العربية، وتتلخّص مساهماتها الأساسية في مرحلة المعالجة الأوّلية، واقتراح طريقةٍ قائمةٍ على تقنيات التّعلُّم العميق، إلى جانب تقنيات زيادة البيانات. تتمثّل المعالجة الأوّلية بتصحيح الالتواء للأسطر وإزالة الفراغات غير الضّرورية فيها. أما بِنية التّعلُّم العميق فتتكون من شبكة عصبية التفافية، ووحدة لاستخلاص السِّمات، إلى جانب ذاكرة ثنائية الاتّجاه طويلة المدى وقصيرة المدى لنمذجة التّتابُع، ومصنِّف مؤقّت لفكّ التّرميز. وأمّا تقنيات زيادة البيانات فيستفاد منها في الصّور المتضمَّنة في قاعدة البيانات في تحسين قدرة النّموذج على تمييز طيْفٍ واسعٍ من الأحرف باللّغة العربية.

وتجدر الإشارة إلى أنّ النّموذج المقترح لديه القدرة على تمييز النّصوص المكتوبة بخطّ اليد دون الحاجة إلى تجزئتها إلى أحرف، مما يساعد في التّغلّب على العديد من المسائل المتعلقة بهذا الموضوع. وقد بينت النّتائج أنّ النّموذج المقترح أثبت عند تجربته على قاعدة البيانات (KHATT) فاعلية جيدة؛ إذ بلغ معدّل خطأ الكلمات 14.55% بينما بلغ معدّل خطأ الأحرف 3.25%.