Journal of Theoretical and Applied Information Technology
31st May 2024. Vol.102. No. 10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ARABIC/ENGLISH MACHINE-PRINTED AND HANDWRITTEN TEXT IDENTIFICATION IN DOCUMENT IMAGES USING IMPRINT TEXTURE AND CNN

AHMAD A. ALZAHRANI
Computer Science and Artificial Intelligence Department, College of Computing, Umm Al-Qura University, Mecca 24382, Saudi Arabia.
E-mail: aalzahrani@uqu.edu.sa

ABSTRACT

The identification of language and writing styles, referred to as script identification, is crucial for automated document images analysis. In Arabic-speaking countries, documents often contain machine-printed and handwritten text in both Arabic and English, which poses a challenge for document image digitization and OCR systems. This paper proposes an image processing with a deep learning-based system that can identify the script type (Arabic or Latin) and its nature (printed machine or handwritten) in document images. Firstly, the system produces an imprint image of the text as input to enhance accuracy. Then using a convolutional neural network (CNN) architecture for feature extraction and classification. The system is trained and evaluated based on benchmark datasets such as the Khatt dataset, the IAM Handwriting Database, the Arabic Sentiment Twitter Corpus dataset, and the LRDE Document Binarization Dataset. The results show that the proposed method significantly improves the identification of text type and style compared to the state-of-the-art techniques.

Keyword: *OCR, Text Identification, Document Images, Arabic Language, CNN.*

1. INTRODUCTION

With the increasing number of document images, document image analysis and recognition (DIAR) has become important [1]–[3]. Practical document images like bank checks, forms, and letters may contain machine-printed and handwritten texts in different languages. The identification of language and written styles in digitized documents, known as script identification, is crucial for automated document analysis [4], [5]. Documents in many Arabic countries may contain machine-printed and handwritten text in Arabic and English languages, which presents a challenge for digitization [6], [7]. Printed text is generally easier to recognize than handwritten text, and separate treatment and recognition algorithms are required for different types of text[8][9], [10]. Accurately identifying handwritten and printed text is essential for recognizing signatures and writers, and for improving document image processing[1], [7], [9].

The integration of OCR systems that can recognize both machine-printed and handwritten text and different languages can improve automatic document processing [11][12]. However, the presence of both printed and handwritten text in the same document image is a significant challenge. To analyze these documents accurately, it is essential to differentiate between machine-printed and handwritten text and classify text languages[4]–[6], [9]. The development of advanced recognition engines for different text types and languages is necessary for efficient digitized document processing in libraries and archives[1], [4], [6]. Previous research has used various methods to identify Arabic and Latin scripts in text blocks, as well as to distinguish between handwritten and printed characters[4], [9], [13]–[15]. These methods rely on local and global features and layout analysis. Some of the techniques proposed include extracting text line features, using statistical and structural features of text lines, Fourier transforms in local areas, edge co-occurrence matrix features, and scale-invariant features as local features. However, these methods have limitations and lack generalizability[16][4], [9], [16]–[18].

The main objective of this research is to propose a deep learning-based system that can

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

identify the script (Arabic or Latin) and its nature (printed machine or handwritten) in document images. The system uses a convolutional neural network (CNN) architecture for feature extraction and classification, including a fully connected network for text nature recognition. Moreover, the proposed system uses an imprint image technique as input instead of traditional slides windows to enhance the system's performance accuracy. This work's main contributions are two-fold: firstly, the proposed preprocessing method to produce an imprinted image of text that enhances deep learning accuracy, and secondly, the use of CNN model for feature extraction and text style and language identification recognition. The proposed system enhances the field of text identification by combining traditional principles used in previous literature with the latest CNN trends. This results in improved automated document analysis applications.

The remainder of this paper is organized as follows: Section 2 presents the literature review, Section 3 explores the proposed method, Section 4 presents the experiments and results, and Section 5 provides a discussion of the results. Finally, Section 6 provides the conclusion of this work.

2. LITERATURE REVIEWS.

According to the literature review, it appears that only a few works have been published on identifying Arabic and Latin scripts in printed and handwritten document images. One such work is by Rouhou et al [18], who proposed a method that uses Hidden Markov Models (HMMs) to identify Arabic and Latin scripts in printed and handwritten documents. Their approach involves training the HMM with a specific dataset for each script type and then extracting features from the test image. These features are then fed into the HMM, which is trained to classify the image as either Arabic or Latin script. The experimental results show that their proposed system can accurately identify the script type and text nature with high precision. The experimental results demonstrate that their proposed system can accurately identify the script type and text nature with high precision.

Another work by [19] proposes an approach to identify Arabic and Latin script types in printed and handwritten documents using Histogram of Oriented Gradients (HOG) descriptors. Their approach applies HOG at the word level based on writing orientation analysis and uses co-occurrence matrices of HOG to consider spatial information between pairs of pixels. A genetic algorithm is applied to select the potential informative feature combinations that maximize the classification accuracy. The output is a relatively short descriptor that provides an effective input to a Bayes-based classifier. The experimental results demonstrate that their proposed system can accurately identify the script type and text nature with high precision. However, variables such as writing style, text size, and font style can affect the accuracy of the system.

In [20], a method was proposed for identifying script type and distinguishing between handwritten and machine-printed text in document images using a Convolutional Neural Network (CNN). The method was trained on a large dataset of document images that contained various script types, such as Chinese, English, Japanese, Korean, or Russian, achieving high accuracy. However, it did not support the Arabic language.

For the classification of printed and handwritten text in a single language, [21] proposed a method that utilized a combination of local and global features, including texture, shape, and density-based features. These features were fed into a Support Vector Machine (SVM) classifier, which achieved high accuracy in distinguishing between handwritten and printed text in doctor's prescriptions. Garlapati et al. [22] also proposed a method to distinguish between handwritten and machine-printed texts in document images using a combination of distinct features in three stages: text localization, feature extraction, and classification. They trained a Support Vector Machine (SVM) classifier using multiple features to classify texts as either handwritten or machine-printed. Malakar et al. [23] proposed a method to classify handwritten and printed word images in a document image using a 6-element feature set for each word image. The features were ranked, and a tree-like classifier was designed based on the ranked features for Latin language.

Hangarge et al. [24] proposed a method for printed and handwritten text classification using texture-based statistical features for South Indian scripts. They extracted statistical texture features for each word image and used them to classify the words using a k-NN classifier. In [25], a method for separating machine-printed and handwritten texts in noisy documents using wavelet transform was proposed. They extracted features from the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

LL sub-band of the wavelet transform using statistical and texture-based methods and trained an SVM classifier to classify the text as either machine-printed or handwritten.

In [26], a method was proposed for distinguishing between handwritten and printed text in document images based on stroke thickness features. They extracted stroke thickness features from text images and trained an SVM classifier, achieving high accuracy even with degraded or noisy text. Finally, [16]proposed a method for classifying printed text and handwritten characters using neural networks. They preprocessed the dataset to extract features such as projections, moments, and Zernike moments, and trained separate neural network models for printed text and handwritten characters using backpropagation and the Levenberg-Marquardt algorithm. The proposed method achieved high accuracy in both printed text and handwritten character classification.

Conducting a literature review on scripts identification in printed and handwritten document images poses several challenges. One of the main issues is the limited availability of literature, particularly for commonly used languages or scripts like Arabic. Additionally, different datasets were used in various studies, making it difficult to generalize methods across different cases and identify the most effective approaches for different languages or types of document images. Furthermore, the field of scripts in printed and handwritten document images is rapidly evolving, with new techniques and approaches being developed and refined. Although previous methods employed traditional techniques such as local and global feature extraction, they neglected to incorporate modern methods like CNN, which were only mentioned in one work and did not include the Arabic language.

3. PROPOSED METHOD

This section presents a proposed system designed to recognize the type of text (Arabic/English) and its style (printed/handwritten) in document images. To achieve that, figure 1 presents the main steps of this work. The proposed method is composed of two main stages: the imprint image generation and the recognition module, which work collaboratively to achieve the overall goal. The module's results categorize the text in document images into English printed, English handwritten, Arabic printed, and Arabic handwritten. An overview of the proposed system is presented in Figure 2, and more details about each stage are provided in the following subsections.

3.1. Textual Imprint Generation

Convolutional neural networks (CNNs) that have fully connected layers require inputs with a fixed size. However, words and text-line images have varying lengths, which cannot be directly used as inputs for CNNs with fully connected layers. Preparing the entire dataset to a uniform size is not feasible because the aspect ratios of text-line images vary greatly. Crude resizing of text-line images can result in significant information loss, which can be detrimental to both script and handwritten/machine-printed identification.

One intuitive solution to the problem of varying text-line lengths is to segment all characters in the text image and warp them to a fixed size for input to the CNN. However, precise character segmentation is difficult because it heavily depends on low-level image processing operations, such as image binarization and edge detection, which can lead to the conglutination of some characters due to multiple resolutions, background clutter, lighting, and noise.

This is particularly true for Latin text images where several characters and strokes may appear in a segmented region. If these regions are resized in a straightforward manner, all characters and strokes are compressed along the vertical direction, resulting in the loss of discriminative information similar to that observed when resizing the entire text image [27]. So, in this stage, the objective is to obtain a clear and significant image that displays the imprint of the targeted text features, which will assist the machine learning model that follows.

This is achieved by ensuring that the features of the character patterns are standardized. This is crucial since having a fixed characteristics dimension allows the images to exhibit the same texture of patterns, which is vital for the neural network to efficiently learn and generalize. Furthermore, utilizing a fixed size reduces the computation cost, memory usage, and processing time necessary for character recognition in the system. This stage also permits the system to treat each textual component of images separately, thereby enhancing the detection performance. This step consists of the following steps:

3.2. Image Binarization

Journal of Theoretical and Applied Information Technology
31st May 2024. Vol.102. No. 10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Binarization of an image is the process of converting an image, usually in grayscale form, into a black and white image. This is common in textual images to distinguish an object (text) from the background. The aim is to remove unwanted information from the image and keep only the required information, which is the text. There are several methods to binarize textual images, but statistical thresholding methods are simpler and

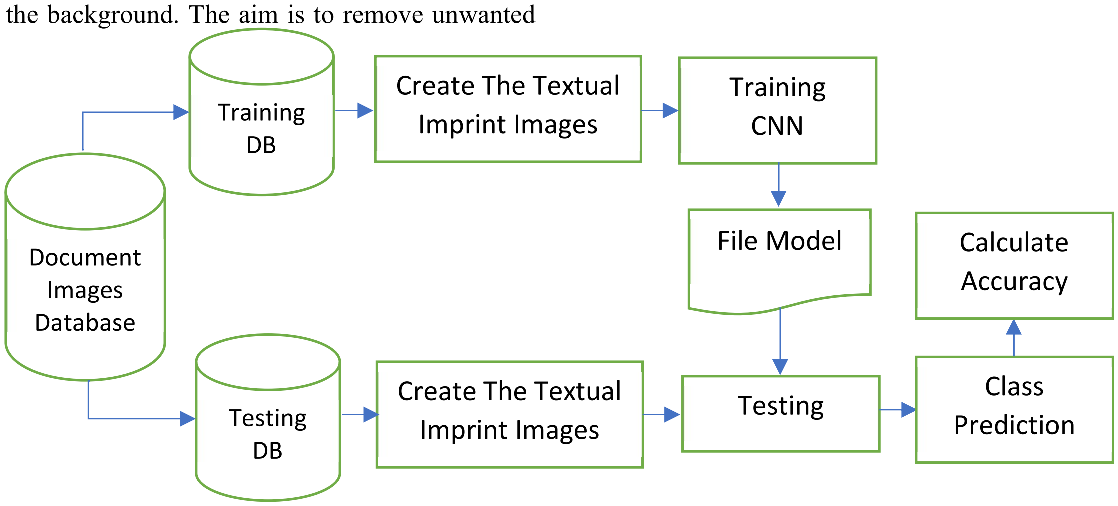
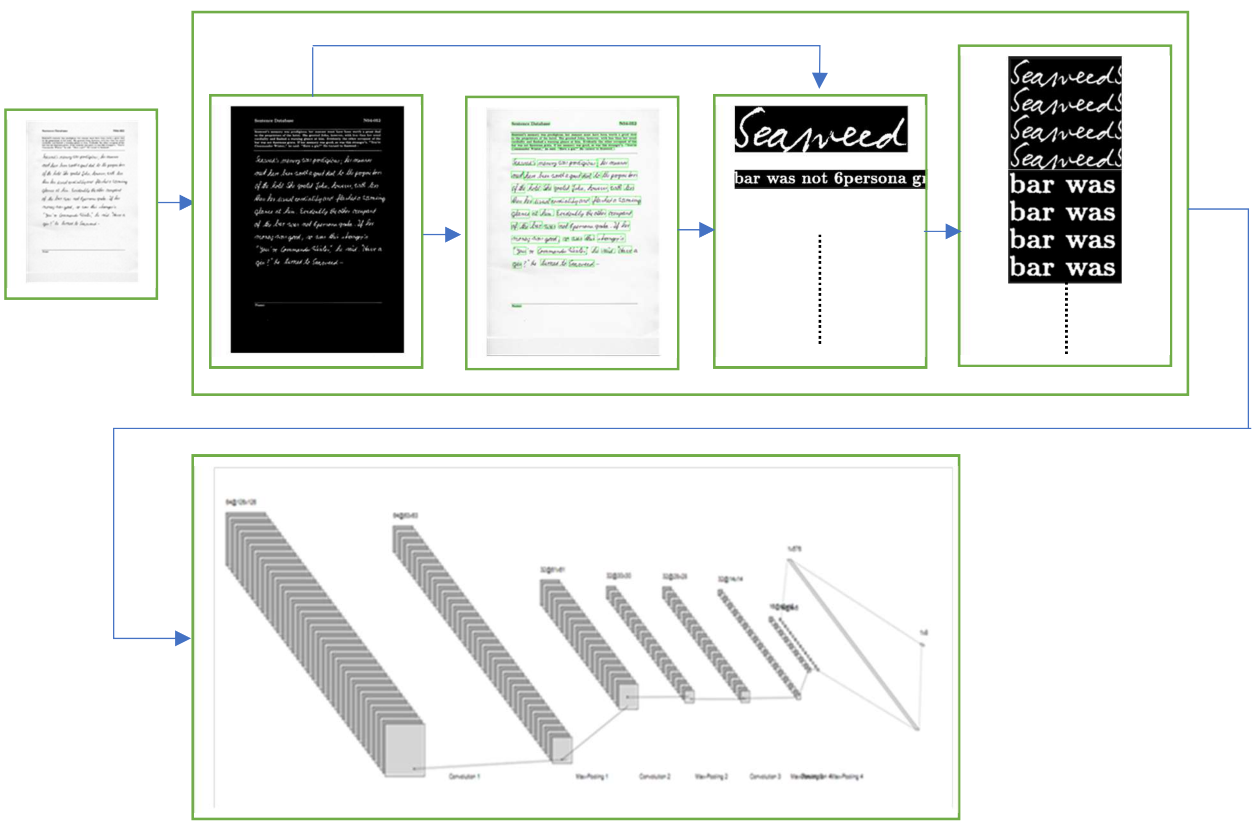


Figure 1. The flowchart of the overall steps of this work



Figuer 2. The architecture of the proposed system

method [29] (equation 1) was used on the grayscale images to separate the text in white from the background in black.

$$T = M_W \frac{M_W^2 \times \sigma}{(m-\sigma)\times(127+\sigma)} \quad (1)$$

Journal of Theoretical and Applied Information Technology
31st May 2024. Vol.102. No. 10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Where, T is the thresholding value between black and white, Mw is the mean value of all pixels in the image, m is the mean value of the window in the image, and σ as the standard deviation of the pixel values in the window. Figure 3(a and b) show an example of the original image and its result of binarization.

3.3. Text Localization

In this stage, the textual patterns present in the image by white pixels are identified and localized. To achieve this, text geometry features are utilized. These features are strokes that are arranged on horizontal lines. In general, the associated text horizontally belongs to the same type. To reduce such number of extracted samples and testing processes later, the morphological close operation is applied[28], [29].

To apply a morphological close operation effectively with different sizes, an appropriate kernel size is needed. To determine this, the first step is to calculate the average width of all individual text components in the image. This value is then used to select an appropriate kernel size K for the morphological close process.

$$\text{Avg}=\frac{\sum C\,(x2-x1)}{N} \qquad (2)$$

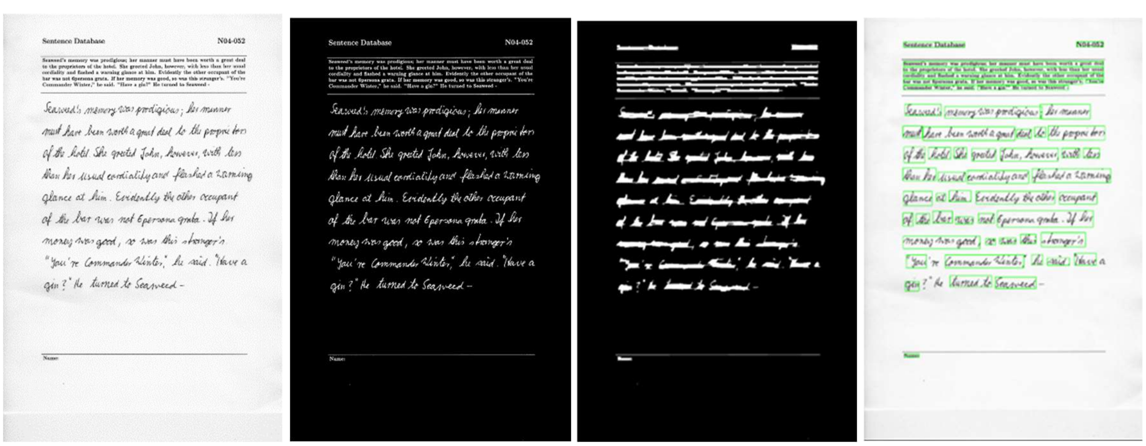
Where C represents a textual pattern component, the width of the component can be identified by its coordinates x1 and x2, and total number of components is denoted by N.

After determining the appropriate kernel size K based on the average width of the individual text components, the next step is to apply a morphological close operation to the image. This process involves two operations: dilation and erosion. The morphological close operation is used to fill gaps and holes between textual patterns horizontally by using a kernel size of (1 x K), K= Avg, and is performed by adding pixels to the image boundaries. This helps to connect associated words of text together (figure 3 (c)) to improve the overall text recognition accuracy.

In the next step, each component region is determined to denote to a segmented image of a binary textual pattern. Which typically consists of separated lines or parts of lines (as shown in Figure 3 (d)). Even if there are overlapping lines, this step is still beneficial for the subsequent steps of the process.

3.4. Textual Imprint Samples Generation

Each segmented component is extracted from the binary image, as illustrated in Figure 4, which shows random selected examples of the results. The segmented parts in a textual image usually belong to the same type of text. Therefore, each textual component is treated as an independent sub-image and is tested separately to determine its type. By treating each component independently, the algorithm can identify the type of text in each component more accurately, as the characteristics of each component may differ from the rest of the image.



(a) (b) (c) (d)

Figure 3. (a) the input gray image, (b) the binarization result, (c) the results after close operation, and (d) is the boundaries of the segmented components of texts

Journal of Theoretical and Applied Information Technology
31st May 2024. Vol.102. No. 10
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 4. an example of the segmented textual components.

This approach allows for a more targeted and precise analysis of the textual content within an image.

After the textual sub-images have been segmented, each sub-image is converted into imprint forms to generate sufficient data for the research. The process for generating these forms follows the following steps:

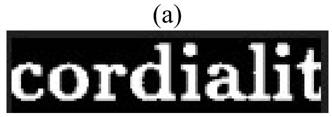
1) First, each component is scaled and resized to a height of 32 pixels, as shown in Figure 5(a).
2) Next, a line is generated from this component with a size of 32 x 128 pixels. If the width of the component is smaller than 128, the component is repeated to fill the entire line. If the width is greater than 128, the line is cut to the appropriate size, as shown in Figure 5 (b).
3) Finally, a textual imprint image with a size of 128 x 128 pixels is generated by vertically iterating the generated line image from the previous step four times, as shown in Figure 5 (c).

This process is repeated for each of the independent textual components to generate an imprint image sample of each candidate part of the text to be tested. By generating image samples for each component, the algorithm can create a comprehensive dataset of textual images that can be used to train and test machine learning models. Figure 6 shows examples of the imprint samples of each language and written style.

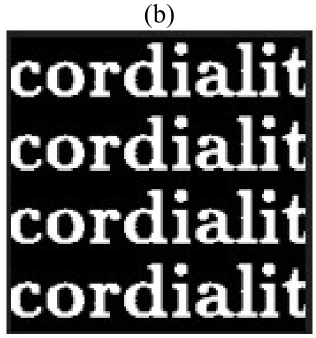
3.5. CNN Architectures

In the second stage, this work proposes a CNN mode fit the imprint images to extract features and identify classes. The convolutional layer employs filters that detect specific features in the input imprint images, producing activation maps that are fed into subsequent layers. The use of multiple filters detecting different features results in a set of activation maps that capture a variety of image characteristics. A pooling layer is inserted between convolutional layers to reduce computation and parameters in the network. This process is repeated through multiple layers to create a deep-learning model capable of extracting increasingly complex features from the input image.

(a)



(b)



ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(c)

Figure 5. (a) the segmented component, (b) a subimago in 32 x 128 size, and (b) the final imprint sample of English printed text in 128 x 128 size

Table 1. The summary of the proposed CNN architecture

Layer	Configurations
Convolution layer 1	Filters=64, stride =1, kernel size=3, activation='relu',
Max-Pooling layer 1	size=2; stride=2 ; padding=1
Convolution layer 2	Filters=32, stride =1, kernel size=3, activation='relu',
Max-Pooling layer 2	size=2; stride=2 ; padding=1
Convolution layer 3	Filters=32, stride =1, kernel size=3, activation='relu',
Max-Pooling layer 3	size=2; stride=2 ; padding=1
Convolution layer 4	Filters=16, stride =1, kernel size=3, activation='relu',
Max-Pooling layer 4	size=2; stride=2 ; padding=1
Fully connected layer 1	Neurons=576
Fully connected layer 2	Neurons=8
sigmoid layer 1	Loss: sigmoid

The proposed CNN architecture consists of four convolutional layers and four max-pooling layers as depicted in Figure 2. it is intended to enhance the recognition module's accuracy and robustness, enabling it to handle a broader range of texture image variations and improve generalization performance. The features extracted from the input image by the convolutional and sub-sampling layers are then passed through fully connected layers for classification. These layers have connections between all the neurons and the activations from the previous layers.

The final fully connected layer, which is the sigmoid layer, outputs the class label of the character corresponding to the input image and provides a set of probabilities for each possible character in the alphabet. In this case, there are 4 possibilities for each input image, including English printed, English handwritten, Arabic printed, and Arabic handwritten. Table 1 provides a summary of the CNN architecture used in this work.

In addition, the paper explores three CNN architectures for the use in script and handwritten/machine-printed identification. The first architecture is small and like LeNet5 [30], with two convolutional layers and two max pooling layers. The second architecture is large, like AlexNet [31], with more convolutional layers than the small architectures, including two inserted between the second and third max pooling layers. The large architecture has a larger receptive field and takes longer to train.

4. EXPERIMENTS AND RESULTS

In this section, several experiments were conducted to evaluate the performance of the proposed method.

4.1. Datasets Description

To cover the various classes included in this system, such as English printed, English handwritten, Arabic printed, and Arabic handwritten, several databases were utilized. These databases include the Khatt dataset [32], the IAM Handwriting Database[33], the Arabic Sentiment Twitter Corpus dataset [34], and the LRDE Document Binarization Dataset[35]. Then, imprint images generation was applied to create a suitable database for training and testing. In total, approximately 8,000 original images were used, and after preparation and imprint image generation, a total of 80,982 imprint image samples were generated, equally divided between English printed, English handwritten, Arabic printed, and Arabic handwritten, with around 20,000 images for each class. Figure 6 shows an example of the imprint images used in training and testing the proposed method.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(a) (b) (c) (d)

Figure 6. examples about imprint images for each of (a) Arabic handwritten, (b) English handwritten, (c) Arabic printed, and (d) English printed.

4.2. Experiment Setup

The proposed system was developed using python and TensorFlow and runs on a computer with an Intel® Core i5 CPU @ 3.70 GHzm, Nvidia GeForce RTX2080 super and 32 GB memory. The performance of each module of the system was assessed by measuring the accuracy of the style recognition rate. This involved calculating the precision by comparing the number of correctly recognized text styles to the actual styles that were supposed to be recognized.

During the training process, 52,638 (66% of the dataset)image data were used as training samples, and the target was to classify the data into four different classes. After completing the training, the accuracy of the training data was evaluated, as well as the accuracy of testing using 28,343 (34% of the dataset) images from the dataset containing the four styles classes. Figure 7 shows the training and testing accuracy and loss against 100 number of epochs.

To show the effectiveness and performance of the proposed models. Table 2 shows the accourcy rate and loss of five independent experiments were conducted, each consisting of training the dataset for 10 epochs followed by testing. To prevent any impact on the results and to reduce overfitting, the dataset was randomly split into two sets in each experiment. The proposed model was evaluated by calculating the average accuracy and loss rate of the test sets in each experiment, which is shown in Table 2. The results indicate that the accuracy rates of the model varied from 98.38% to 98.1% across the five experiments.

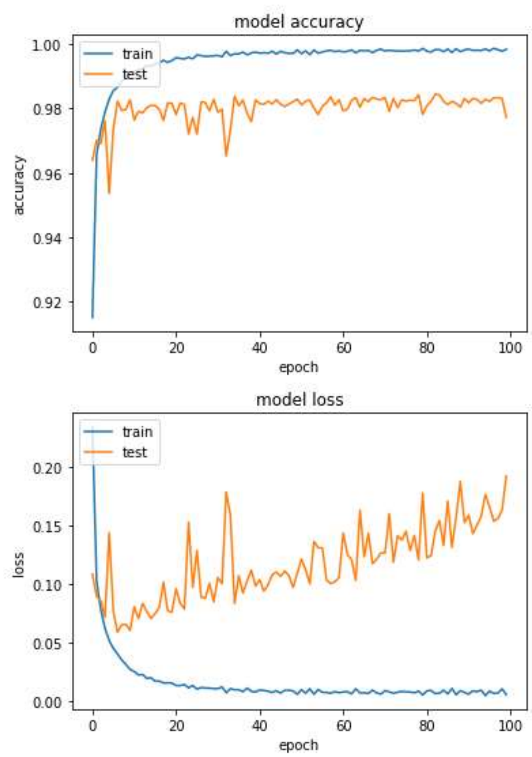


Figure 7. Proposed model's graph of training and testing (a) accuracy and (b) loss against number of epochs.

The model's performance was consistent across different deviated datasets, with an average

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

accuracy rate of 98.21 and a standard deviation of 0.126.

Table 2. The accuracy and loss rates of the proposed CNN models based on the test dataset.

Experiment	**1**	**2**	**3**	**4**	**5**	***Average***	***StD***
Accuracy	98.38	98.30	98.12	98.1	98.13	98.21	0.126
Loss of	0.0520	0.0578	0.0639	0.0603	0.0630	0.059	0.0048

4.3. Performance Analysis

Another experiment was conducted to compare the performance of different deep learning-based architectures in recognizing characters. The architectures evaluated included LeNet5[30] and AlexNet[31], as well as a similar method of Arabic/Latin Handwritten/Printed Identification System. The selected methods were A HMM-Based [18], LBP+SVM [20], EDMS [36]. In addition, the effectiveness of proposed imprint input preparation is compared with the traditional inputs based on slide windows. The inputs are resized to 128 ×128 to fit the input CNN methods.

Based on the results shown in Table 3, the proposed system achieved a higher identification recognition accuracy of 98.38% compared to LeNet5 and AlexNet CNN by 96.13 and 97.32 respectively. In general, CNN-based methods outperformed traditional methods that rely on traditional feature extraction and recognition stages. Furthermore, the impact of the preprocessing stage in producing imprint images for inputs was compared to the usual slide window. The accuracy improved from 93.98% to 98.38% when the proposed imprint inputs were used. The positive effects of this method were observed in all the involved methods CNN.

Table 3. the script identification accuracy of involved methods with each of slide windows and imprint images inputs.

Method	Accuracy (Imprint images)	Accuracy (Slide window)
AHMM-Based [18]	-	91.56
LBP+SVM [20]	47.04	85.5
EDMS+NN[36]	56.42	73.44
EDMS+SVM	47.03	47.56
LeNet5[30]	96.13	91.62
AlexNet[31]	97.32	95.82
Proposed	**98.38**	93.98

5. DISCUSSION

The contribution of this work is the development of a CNN model for accurately classifying English/Arabic text in machine-printed and handwritten styles. The model utilizes image pre-processing techniques, generating an imprint texture block to represent text features. The proposed CNN models have simple architectures with varying numbers of layers and kernel sizes, aimed at improving performance.

Results demonstrate that the proposed model achieved the best performance among all methods, with an accuracy of 98.38% when combined with the proposed imprint input image. The imprint input improved the results of all machine learning methods and was more effective than the traditional slide window technique. The proposed CNN model is designed to fit textual imprint images and outperforms well-known KNN models such as LeNet5[30] and AlexNet[31].

For future work, other input forms could be explored to improve the performance of machine learning algorithms. The efficiency of the proposed model for other languages could be investigated, and different optimizers could be compared.

6. CONCLUSION

In this work, we propose a system that can identify Arabic or Latin text and distinguish between printed machine and handwritten nature in document images. Firstly, an imprint image of the text is produced and used as input to a proposed convolutional neural network (CNN) for feature extraction and classification. The system is trained and evaluated on various datasets, including the Khatt dataset, the IAM Handwriting Database, the Arabic Sentiment Twitter Corpus dataset, and the LRDE Document Binarization Dataset. The results show that the proposed method significantly improves the identification of text type and style, achieving a 98.38% accuracy rate. The use of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

imprint input improves the results of all machine learning methods and is more effective than the traditional slide window technique. The proposed CNN model is designed to fit textual imprint images and outperforms well-known KNN models such as LeNet5 and AlexNet.

REFERENCES

[1] S. Patra and J. Sahoo, "A Literature Review on Digitization in Libraries and Digital Libraries," *Preserv. Digit. Technol. Cult.*, vol. 51, no. 1, pp. 17–26, 2022.

[2] B. M. A. Bataineh and M. K. Y. Shambour, "A robust algorithm for emoji detection in smartphone screenshot images," *J. ICT Res. Appl.*, vol. 13, no. 3, pp. 192–212, 2019, doi: 10.5614/itbj.ict.res.appl.2019.13.3.2.

[3] T. Abu-Ain, S. N. H. S. Abdullah, B. Bataineh, W. Abu-Ain, and K. Omar, "Text normalization framework for handwritten cursive languages by detection and straightness the writing baseline," *Procedia Technol.*, vol. 11, pp. 666–671, 2013.

[4] K. Ubul, G. Tursun, A. Aysa, D. Impedovo, G. Pirlo, and T. Yibulayin, "Script identification of multi-script documents: a survey," *IEEE access*, vol. 5, pp. 6546–6559, 2017.

[5] K. Yang, J. Yi, A. Chen, J. Liu, W. Chen, and Z. Jin, "ConvPatchTrans: A script identification network with global and local semantics deeply integrated," *Eng. Appl. Artif. Intell.*, vol. 113, p. 104916, 2022.

[6] S. Bin Ahmed, S. Naz, M. I. Razzak, and R. B. Yusof, "A novel dataset for English-Arabic scene text recognition (EASTR)-42K and its evaluation using invariant feature extraction on detected extremal regions," *IEEE access*, vol. 7, pp. 19801–19820, 2019.

[7] A. Y. Muaad *et al.*, "Arabic document classification: performance investigation of preprocessing and representation techniques," *Math. Probl. Eng.*, vol. 2022, pp. 1–16, 2022.

[8] A. Gaygole and D. Rojatkar, "A review on methods of script identification for printed and handwritten documents," *2019 Innov. Power Adv. Comput. Technol.*, vol. 1, pp. 1–6, 2019.

[9] D. Sinwar, V. S. Dhaka, N. Pradhan, and S. Pandey, "Offline script recognition from handwritten and printed multilingual documents: a survey," *Int. J. Doc. Anal. Recognit.*, vol. 24, pp. 97–121, 2021.

[10] K. Hemalatha, C. Lakshmi, N. C. P. Raju, S. I. Zabiulla, D. A. Kumar, and S. A. Kumar, "HandWritten Text Recognition Using Machine Learning," *J. Eng. Sci.*, vol. 14, no. 02, 2023.

[11] B. Bataineh, "A Printed PAW Image Database of Arabic Language for Document Analysis and Recognition.," *J. ICT Res. Appl.*, vol. 11, no. 2, 2017.

[12] B. Bataineh, S. N. H. S. Abdullah, K. Omar, and M. Faidzul, "Adaptive thresholding methods for documents image binarization," in *Pattern Recognition: Third Mexican Conference, MCPR 2011, Cancun, Mexico, June 29-July 2, 2011. Proceedings 3*, 2011, pp. 230–239.

[13] I. Guellil, H. Saâdane, F. Azouaou, B. Gueni, and D. Nouvel, "Arabic natural language processing: An overview," *J. King Saud Univ. Inf. Sci.*, vol. 33, no. 5, pp. 497–507, 2021.

[14] K. O. M. Aarif and P. Sivakumar, "Cursive script identification using Gabor features and SVM classifier," *Int. J. Comput. Aided Eng. Technol.*, vol. 12, no. 3, pp. 328–335, 2020.

[15] R. Aggarwal, S. Pandey, A. K. Tiwari, and G. Harit, "Survey of mathematical expression recognition for printed and handwritten documents," *IETE Tech. Rev.*, vol. 39, no. 6, pp. 1245–1253, 2022.

[16] K. B. Neelima and S. Arulselvi, "Classification of printed text and handwritten characters with neural networks," *J Crit Rev*, vol. 7, pp. 134–139, 2020.

[17] K. Zagoris, I. Pratikakis, A. Antonacopoulos, B. Gatos, and N. Papamarkos, "Handwritten and machine printed text separation in document images using the bag of visual words paradigm," in *2012 International Conference on Frontiers in Handwriting Recognition*, 2012, pp. 103–108.

[18] A. Cheikh Rouhou, Y. Kessentini, and S. Kanoun, "HMM Based Keyword Spotting System in Printed/Handwritten Arabic/Latin Documents with Identification Stage," in *Image Analysis and Recognition: 16th International Conference, ICIAR 2019, Waterloo, ON, Canada, August 27–29, 2019, Proceedings, Part I 16*, 2019, pp. 309–320.

[19] A. Saidani, A. Kacem Echi, and A. Belaid, "Arabic/Latin and machine-printed/handwritten word discrimination using HOG-based shape descriptor," *ELCVIA Electron. Lett. Comput. Vis. image Anal.*, pp. 1–23, 2015.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[20] G. G. Rajput and S. B. Ummapure, "Script Identification from Handwritten document Images Using LBP Technique at Block level," in *2019 International Conference on Data Science and Communication (IconDSC)*, 2019, pp. 1–6.

[21] D. Dhar, A. Garain, P. K. Singh, and R. Sarkar, "HP_DocPres: A method for classifying printed and handwritten texts in doctor's prescription," *Multimed. Tools Appl.*, vol. 80, pp. 9779–9812, 2021.

[22] B. M. Garlapati and S. R. Chalamala, "A system for handwritten and printed text classification," in *2017 UKSim-AMSS 19th International Conference on Computer Modelling & Simulation (UKSim)*, 2017, pp. 50–54.

[23] S. Malakar, R. K. Das, R. Sarkar, S. Basu, and M. Nasipuri, "Handwritten and printed word identification using gray-scale feature vector and decision tree classifier," *Procedia Technol.*, vol. 10, pp. 831–839, 2013.

[24] N. Dutly, F. Slimane, and R. Ingold, "PHTI-WS: a printed and handwritten text identification web service based on FCN and CRF post-processing," in *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, 2019, vol. 2, pp. 20–25.

[25] P. Sahare and S. B. Dhok, "Separation of machine-printed and handwritten texts in noisy documents using wavelet transform," *IETE Tech. Rev.*, vol. 36, no. 4, pp. 341–361, 2019.

[26] H. Ding, H. Wu, J. Wang, and X. Zhang, "Handwritten and printed text distinction by using stroke thickness features," in *Seventh International Conference on Electronics and Information Engineering*, 2017, vol. 10322, pp. 295–301.

[27] Z. Feng, Z. Yang, L. Jin, S. Huang, and J. Sun, "Robust shared feature learning for script and handwritten/machine-printed identification," *Pattern Recognit. Lett.*, vol. 100, pp. 6–13, 2017.

[28] T. Abu-Ain, S. N. H. Sheikh Abdullah, B. Bataineh, K. Omar, and A. Abu-Ein, "A novel baseline detection method of handwritten Arabic-script documents based on sub-words," in *Soft Computing Applications and Intelligent Systems: Second International Multi-Conference on Artificial Intelligence Technology, M-CAIT 2013, Shah Alam, August 28-29, 2013. Proceedings*, 2013, pp. 67–77.

[29] T. Abu-Ain, S. N. H. S. Abdullah, B. Bataineh, and K. Omar, "A Fast and Efficient Thinning Algorithm for Binary Images.," *J. ICT Res. Appl.*, vol. 7, no. 3, 2013.

[30] G. Wei, G. Li, J. Zhao, and A. He, "Development of a LeNet-5 gas identification CNN structure for electronic noses," *Sensors*, vol. 19, no. 1, p. 217, 2019.

[31] M. Z. Alom *et al.*, "The history began from alexnet: A comprehensive survey on deep learning approaches," *arXiv Prepr. arXiv1803.01164*, 2018.

[32] S. A. Mahmoud *et al.*, "Khatt: Arabic offline handwritten text database," in *2012 International conference on frontiers in handwriting recognition*, 2012, pp. 449–454.

[33] U.-V. Marti and H. Bunke, "The IAM-database: an English sentence database for offline handwriting recognition," *Int. J. Doc. Anal. Recognit.*, vol. 5, pp. 39–46, 2002.

[34] M. Nabil, M. Aly, and A. Atiya, "Astd: Arabic sentiment tweets dataset," in *Proceedings of the 2015 conference on empirical methods in natural language processing*, 2015, pp. 2515–2519.

[35] X. Peng, H. Cao, and P. Natarajan, "Using convolutional encoder-decoder for document image binarization," in *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, 2017, vol. 1, pp. 708–713.

[36] B. Bataineh, S. N. H. S. Abdullah, and K. Omar, "A novel statistical feature extraction method for textual images: Optical font recognition," *Expert Syst. Appl.*, vol. 39, no. 5, pp. 5470–5477, 2012.