WSNet – Convolutional Neural Network-based Word Spotting for Arabic and English Handwritten Documents

Hanadi Hassen Mohammed [1], Nandhini Subramanian [1],
Somaya Al-Maadeed [1], Ahmed Bouridane [2]

[1] *Department of Computer Science and Engineering, College of Engineering,*
Qatar University Al Jamiaa St, Doha, Qatar

[2] *Centre for Data Analytics and Cybersecurity, University of Sharjah,*
United Arab Emirates

Abstract **– This paper proposes a new convolutional neural network architecture to tackle the problem of word spotting in handwritten documents. A Deep learning approach using a novel Convolutional Neural Network is developed for the recognition of the words in historical handwritten documents. This includes a pre-processing step to re-size all the images to a fixed size. These images are then fed to the CNN for training. The proposed network shows promising results for both Arabic and English and both modern and historical documents. Four datasets – IFN/ENIT, Visual Media Lab – Historical Documents (VML-HD), George Washington and IAM datasets – have been used for evaluation. It is observed that the mean average precision for the George Washington dataset is 99.6%, outperforming other state-of-the-art methods. Historical documents in Arabic are known for being complex to work with; this model shows good results for the Arabic datasets, as well. This indicates that the architecture is also able to generalize well to other languages.**

DOI: 10.18421/TEM111-33
https://doi.org/10.18421/TEM111-33

Corresponding author: Hanadi Hassen Mohammed, *Department of Computer Science and Engineering, College of Engineering, Qatar University Al Jamiaa St, Doha, Qatar.*
Email: hm1409611@qu.edu.qa

Received: 02 October 2021.
Revised: 26 January 2022.
Accepted: 02 February 2022.
Published: 28 February 2022.

Keywords **–Word spotting, Deep learning, Word recognition, Arabic word spotting**

1. Introduction

Word spotting, which is defined as the process of identifying all the instances of a word found in an entire document, has extensive applications, such as indexing, annotation, subword recognition, and deciphering document images, especially handwritten ancient, historical and modern documents. Historical documents contain valuable information preserving the details of the culture of the era of their creation. At times, analyzing these handwritten documents can become difficult because of various factors. The query word used for searching in the word spotting methods can be either an image (query-by- example) or text string (query-by-string). Various methods have been proposed over the years for performing word spotting, namely, feature engineering, feature engineering along with machine learning and feature engineering with deep learning. In this paper, a new and efficient approach to word spotting in handwritten historical documents is presented using deep learning methods. Such methods have recently been used in a wide range of applications and have been proven to attain higher accuracy and efficiency than other techniques. A deep learning architecture using convolutional neural networks (CNN) is described in this paper. The CNN architecture uses convolution filters as the feature engineering method to learn the features from document images. Since this model mainly aims at word spotting in manuscripts, it will be referred to as WSNet throughout the paper.

One of the major disadvantages of using a deep learning model is the necessity of a larger labeled training dataset. This paper discusses in detail various data augmentation techniques used to overcome the need for larger datasets. The model is trained using both English and Arabic datasets to evaluate its versatility and generalization. This paper applies a segmentation-based method that uses the ground truth provided along with the dataset to segment the entire document image into single word images with the transcription of the respective word as the label. Word images belonging to the same class are organized in a single folder with the class name used as the name of the folder. Although the training process of deep learning models may be computationally intensive, it can be improved by selecting proper parameter values in the settings.

2. Related Works

Word spotting can be performed without recognizing words inside documents via image matching techniques. Feature extraction is an indispensable step of this approach to word spotting. Various feature extraction techniques have been proposed in the literature on automatic word spotting. One such technique is the local binary pattern (LBP) method that is known for its simplicity and discriminative power. The LBP method turns the high- dimensional space based on pixel intensities into a low-dimensional space that only encodes the relative intensity values. In [1], the LBP combined with spatial sampling showed stability of the LBP under handwriting deformations. Another method has been proposed by [2] to tackle the problem of handwriting deformation. The researchers proposed a bidirectional Dynamic Time Wrapping (DTW) strategy. The DTW is used to measure the similarity of two time series and has been widely applied in speech recognition. A bidirectional Dynamic Time Warping is used in [2] to measure the distance between two words, and the histogram of gradient (HoG) descriptors is used to describe the local information of a word’s image. Another example of word spotting based on feature engineering is shown in [3]. The authors used an unsupervised hierarchical representation of handwriting based on spherical K-means to extract discriminative features from document images. Those extracted features were then compressed, and a sliding window was used to detect regions that matched the query image’s representation.

Word spotting can be performed using machine learning techniques. In this case, the query words must be known to train the model. Support vector machines (SVM) and hidden Markov models (HMM) are widely used for the task of word spotting. The authors [4] applied the SVM to training a model for Urdu words extracted from documents using a sliding window that combined the neighboring connected components. The extracted words were represented using gradient and profile-based features. The reported performance of the SVM on the CENPARMI Urdu Database was 50% precision at a recall rate of 70.1%. Instead of segmenting documents into pieces of words, text lines can be used to train a machine learning model. Fischer et al. [5] proposed an HMM model for learning the whole text lines represented by a set of features extracted using a sliding window. The reported precision was 31.5% on the GW database. Later, in [6], the researchers improved their HMM-based word spotting model by utilizing n-grams in the model.

PHOCNet [7] is a deep learning convolutional neural network that uses a representation of labels as a pyramidal histogram of characters (PHOC). This multilabel model uses the spatial pyramid pooling (SPP) layer before the flattened layer, which facilitates processing of images of arbitrary sizes. Since the labels are represented as their PHOC attributes, the model can work with both query-by-example and query-by-string scenarios. TPP-PHOCNet [8] is an improved PHOCNet using the temporal pyramid pooling (TPP) layer designed specifically to input word images for word spotting research. In this paper, other string word representations such as the discrete cosine transformation of words (DCToW), the spatial pyramid of characters (SPOC) and various loss functions suitable for these string embedding are discussed.

The HWNet [9] is used to detect the similarity in two different documents written by different authors using the convolutional network architecture. The architecture is first pre-trained on the synthetic dataset HW-SYNTH constructed from 750 different publicly available hand- written fonts. The weights are then transferred while learning with the handwritten datasets. The HWNet v2 [10], an improved version of HWNet presented in [9], is an adaptation of the ResNet-34 architecture with the region of interest (ROI) pooling layers that help in reading images of variable sizes. In the work, a synthetic dataset IIT-HWS was created from the publicly available handwritten fonts. This dataset consists of approximately 1 million word images and hence replaces the need for data augmentation methods.

In [11], the authors use the embedded attributes of both the word image and label text to learn the subspace where both the word image and label are found. The HWNet architecture discussed above is used in the method. The HWNet has been improved in [5] by using an end-to-end embedding framework. Here, a synthetic dataset has again been used to improve the performance of the model.

Authors of [12] used a new architecture to relate the frequencies of N-grams such as unigrams, bigrams and trigrams of the word label and parts of it, and subsequently performed the canonical correlation analysis to match the words and the predictions. The authors used both handwritten and printed datasets to prove the potential of the designed network. An attribute vector was formed from the text label and the corresponding images, and was then fed to the network for training. In [13] and [14], recurrent neural networks were used for word recognition.

In [15] – [16], HMM-based word recognition was performed. Image were skeletonized first, and then each pixel was categorized as either horizontal or vertical. These were then coded into numeric vectors in [17]. A fully connected deep learning model was used in [18] on Arabic datasets. The IFN/ENIT datasets were used for evaluation in the above-mentioned studies.

A detailed survey answering if the performance is better with increasing architecture of the CNN, how deep the CNN has to be for word spotting can be seen in [19]. The authors in [19] have considered the recent TPP-PHOCNet for comparison between residual networks and LeNet architectures. A convolutional Siamese network consisting of two convolutional networks to study the similarity between two word images has been proposed by the authors in [20]. The VML-HD dataset has been used in [20] for evaluation purposes.

3. Proposed Method

Architecture Overview

Before analyzing the architecture of the deep learning model proposed in this paper, a brief description of all the layers used in the deep learning model is provided. The layers used in any deep learning model can be divided into three types: convolutional, pooling and fully connected layers. Convolution can be defined as the process of masking the image with a small matrix usually of size 3x3 or 5x5 to produce the resulting image that can be used for computations. Convolutional layers are used in deep learning models to learn important features present in the image. At each layer in the model, a particular feature of the image is learned. Though the size of the input image is small, the image resulting from the convolutional layer can be quite large, which makes it difficult to process. Pooling layers are used at this stage to reduce the overall size of the image resulting from the convolutional layer without losing any of the important information in the data. Various types of pooling layers exist. However, the max pooling layer is used in this paper because of its wide usage. In the pooling layer, a square filter of defined size is moved over the sliding window in the image, and the maximum value in each window is preserved.

Fully connected layers are used to learn a nonlinear combination of features present in the input. The outputs of the previous layers are flattened and then fed to the fully connected layers with an activation function. A brief summary is that convolutional layers are used to extract the important features present in the image, and fully connected layers are used in classifying the features extracted to their respective classes.

The overall architectural view of the model used in this paper is shown in Figure 1. As Figure 1. shows, the number of filters used increases with each layer. The number of filters increases from 64 to 1024 with each layer. The filter size used in each convolutional layer is a 5x5 matrix without any stride and padding. A max pooling layer of size 2x2 is used with a stride of 2 without any padding. Three fully connected layers are used with sigmoid activation in the last layer for multiclass classification. In total, 13 layers are used in the entire architecture. A batch normalization is performed after each layer to normalize the input to the next layer to keep the mean close to 0 and standard deviation to 1.

The transcription of the word's image from the ground truth is used for the training label process. Though most recent studies related to word spotting have used label attributes constructed from the label using N-grams present in the label, in this paper labels are used without constructing the label attributes. Building the label attributes – an array of 0's and 1's with size varying up to 604 depending on the number of N-grams used – takes up a significant amount of space.

Training the convolutional neural network with label attributes can take a long time, and the selection of a good optimizer can become tricky. Additionally, special attention is required and an in-depth knowledge of the language used is essential in building the PHOC attributes when analyzing databases of texts written in languages other than English, e.g., Arabic, Chinese, etc. Hence, labels rather than label attributes are used in this architecture.

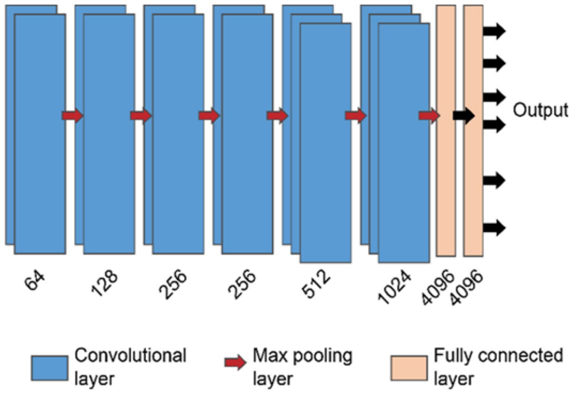

Figure 1. Overall architecture of the convolutional neural network used

A good optimizer has to be selected to reduce the training time of the model. The Adam optimizer is used in training, as it reduces the training time. The Adam optimizer, which is an extension of the stochastic gradient descent (SGD) implementation, is a modern optimizer. Adam optimizers are efficient compared to other optimizers used in solving practical deep learning problems.

Based on various experiments and a careful analysis, the computational parameters described below have been selected. The combination of one forward- and one back-propagation in a neural network is called an epoch. In this paper, the number of epochs is set to 6, and the batch size is 32. The initial learning rate is chosen to be 1e-4 and decay is set to 5*1e-5.

Computational Parameters

The laptop used for this experiment was a GT75VR 7RF system with Intel(R) Core (TM) i7-7820HK CPU operating at 2.90GHZ (2901 MHz) with 4 cores and 8 logical processors. An NVIDIA Geforce GTX 1080 graphics card was used to reduce the training time. The total training time for each dataset was approximately 5 to 7 hours.

Training and Testing Setup

This architecture was mainly developed to analyze historical handwritten datasets. Fully connected layers in the CNN require all the input vectors to be of the same size. Hence, the input images have to be of the same size after segmentation using the ground truth. A basic preprocessing of all the word images is performed by resizing the input images to the same width and height. Selection of proper width and height dimensions plays a vital role in increasing the training and testing accuracy. It is observed that the testing accuracy increases by 2.4% for IFN datasets as a result of choosing the proper image size.

Historical datasets are inherently small in size making the training process quite a tricky task; e.g., the George Washington dataset has a total of 4890 word images. Additionally, these datasets do not have a good diversity of images, e.g., images of the same word with different lighting, noise, angles and rotations for each class. The availability of very little amount of labeled training data and the lack of diversity makes the training of the deep convolutional neural network very difficult. To overcome these problems, data augmentation methods are used.

Such methods can be used to increase diversity and the number of images in each class. Some of the data augmentation methods used are scaling, rotation, adding noise, flipping and perspective transformation. Images are resized to a fixed size of 100x100. No padding is used when resizing the image. The resized images are then augmented using perspective transformation, which after a wide range of experiments gave better results than other augmentation techniques. Affine transformation with random limits between 0.8 and 1.1 is considered for data augmentation.

One more disadvantage in using the historical datasets is that there are some classes with only one image instance. Using these data as such for training the CNN can result in overfitting or underfitting. All the classes available in the dataset have to be balanced to obtain a better generalizing trained model. Original images in the dataset are augmented to obtain the total of 5, 00,000 images. All of the classes are balanced, as such images being present in all the classes result in classes having the same number of images. If the training/testing split is specified in the dataset, the specified partition is used. If unspecified, the data is divided into two sets: 80% is used as training data, and 20% is retained for testing. The training data is then divided into two sets – training and validation sets – for fine-tuning the model. Figure 2. showcases the training and validation accuracy, training and validation loss for all the four datasets used.

4. Experimental Setup

Datasets Used

This paper targets the historical handwritten documents; however, modern handwritten documents are also considered to evaluate the model's versatility and adaptability. There is very little research on Arabic manuscripts because of the difficulty in analyzing such scripts. Arabic scripts have curve-like letters; the text direction is from right to left; and text has superscripts and postscripts to help in pronunciation. English and Arabic datasets are both considered. No historical datasets containing Arabic manuscripts with proper ground truth labeling are publicly available, and hence two modern datasets have been considered. Figure 3. shows samples from each of the datasets used on the word level. Each dataset has segmented word images and the ground truth labels for word images. The model is trained on four different datasets that are described in detail below.

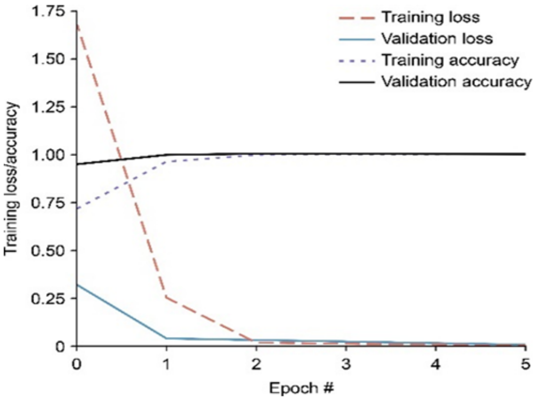

Figure 2. Training accuracy, training loss, validation accuracy and loss on the George Washington dataset

The **IFN/ENIT dataset** is a modern Arabic database containing zip codes and names of cities written by different people. Though not historical, IFN/ENIT dataset is handwritten and poses the same challenges as those of historical handwritten datasets. There are seven different subsets in the dataset, from which subsets a, b, c are selected for training, and subset d is used for testing.

The **VML_HD dataset** is also a modern Arabic dataset containing handwriting. It consists of five books by different authors dating back to 1088-1451. This database is widely used for subword recognition problems. Since there are very few databases of Arabic manuscripts available, and this database has not been used in word spotting research, we have selected it. It is noted that this dataset has been used for subword recognition.

The **George Washington dataset** is a historical English dataset containing letters written by George Washington and his associates during the 18th century. This is a good example of data of a single writer and is used widely in word spotting research. The entire dataset contains approximately 20 pages of letters and the ground truth. After segmentation, there are 4850 word images present in the dataset. The training and testing split is considered as in [7] to make it easier to compare results.

The **IAM dataset** is also a modern English dataset suitable for the multiple authors' scenario. The database contains pages written by 657 authors. The entire dataset comprises approximately 115 320 words. Images of words of various sizes written by different authors are present. The official training and testing split is considered.

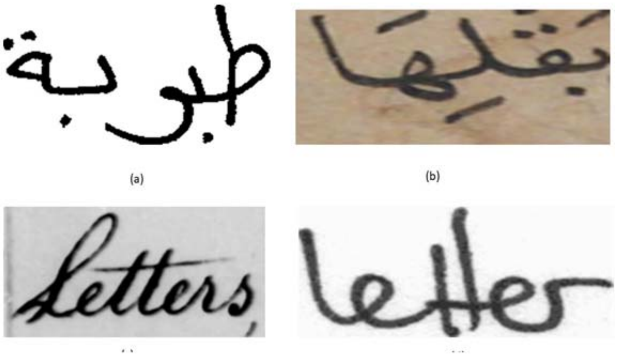

Figure 3. Examples from IFN/ENIT (a), VML-HD (b), GW (c) and IAM (d) datasets

Evaluation Metrics

The mean average precision is calculated for predictions made for the testing set. The precision of each prediction for the testing set obtained with the trained model is calculated. The formula for the precision of the prediction made is given in 1. It is observed that the sum of TP and FP is equal to the total number of images used in the testing set. Then, the average precision at rank k is calculated using (2).

$$P = \frac{TP}{TP+FP} \qquad (1)$$

where P is the precision of the prediction made, TP is the number of true positives among the predictions made, and FP is the number of false positives among the predictions made.

$$AP = \frac{\sum_{q=0}^{k} P_q*rel(q)}{k} \qquad (2)$$

where AP is the average precision, k is the rank at which average precision has to be calculated, Pq is the precision at q.

Finally, the mean average precision of the trained model is calculated by

$$mAP = \frac{\sum_{q=0}^{n} AP_q}{n} \qquad (3)$$

where mAP is the mean average precision, and n is the total number of images in the testing set, and AP_q is the average precision at q.

To compare the results obtained by the WSNet with those of other related studies, more than one evaluation methods is used. In addition to mean average precision, the word error rate and the character error rate are calculated. The WER can be defined as a measure of how significantly the predicted label differs from the original ground truth label.

Suppose that S is a substitution, D is a deletion made, and I is an insertion made so that the predicted label equals the ground truth label; then,

$$WER = \sum_{i=0}^{N} \frac{D_{i+}S_i+I_i}{GT_i} \qquad (4)$$

Where, N is the number of predicted words, and GT is the ground truth label.

Results and Discussion

The proposed method has been compared with other existing methods available in the word spotting research field. For ease of comparison, as each of the papers uses a different evaluation method, mAP, WER and CER are all calculated for the proposed method. In Table 1., the results produced by the proposed network architecture on the IFN/ENIT dataset are shown. Research on this Arabic dataset is scarce, so only mAP values of the results are shown and compared with those of the other methods applied to this dataset. The results are obtained by using subsets a, b and c for training and subset d for testing. It can be seen that our method has similar proposal accuracy to the existing best methods as in [18], and it outperforms all the methods as in [15].

Table 1. Results of the proposed method on the IFN/ENIT Dataset and a comparison of results to those of other methods

Method	mAP (%)
Pechwitz and Maergner [15]	89.74
Alabodi and Li [17]	93.3
Dreuw et al. [21]	96.5
PHOCNET [7]	96.11
Azeem and Ahmed [16]	97.7
Ahmad et al. [22]	97.22
Stahlberg and Vogel [18]	97.6
Proposed method	**97.9**

Table 2. shows the results on the VML-HD dataset. For VML_HD dataset, comparison has been done against the convolutional Siamese network where two convolutional networks are used to study the similarity between the query and the document word images. It is observed that the proposed network also obtains good results on the datasets with more than 2000 labels. From Table 2., it is clear that the proposed method has significant improvement in performance when compared to other methods [20]. Both IFN/ENIT and VML-HD are modern Arabic datasets and the proposed model has good performance results on them too, showing the robustness, versatility and its ability to generalize.

Table 3. represents the results, as measured by mAP, on the George Washington database and compares them with results of other studies that report values of mAP, WER and CER. The proposed method works well with datasets containing samples from multiple authors, for example, IAM has examples from many authors and the model performed effectively with IAM. The model architecture performed better with both English and Arabic data.

Table 4. shows the results on the IAM dataset, and Table 5. compares the overall results on all the datasets used with those of other methods in terms of mAP, WER and CER. The results on the GW dataset have outperformed other methods significantly. Some of the observations that can be made from the results in Table 4. are listed below. Data augmentation works well for handwritten datasets when the number of original samples are very less. The GW dataset has only 5000 samples and after augmenting the samples, the mAP reached 99.6. The proposed method has outperformed all the existing methods [7, 9, and 10] for both the GW and IAM datasets.

Table 2. Results of the proposed mrthod on the VML-HD dataset

Method	mAP(%)
Barakat et al. [20]	61.00
Proposed method	**71.26**

In summary, Table 5. represents the results obtained by the proposed WSNet on all the datasets and compares such results with those of other methods of word spotting. Diagrammatic representation of Table 5. can be seen in Figure 4. Figure 4. compares the results on all the four datasets used by the proposed method with all other methods in the literature.

Also, the proposed model works well with multiple classes like in all the datasets used. The labels were converted to categorical matrix and used as such without any modifications as in PHOCNet [7]. This reduces the overhead on converting the label into PHOC during training and converting the PHOC back to label while testing the models for predictions.

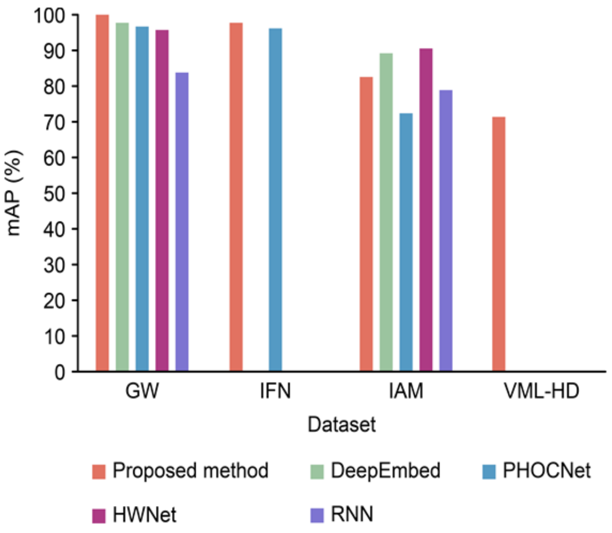

Figure 4. Test results obtained by the method proposed in this paper on four different datasets compared with those of other methods in related fields

Table 3. Results of the Proposed Method on the GW Dataset and a Comparison of the Results to those of other Methods

Method	mAP(%)	WER	CER
RNN [13]	84.0	-	-
CNN-RNN [14]	-	12.98	4.29
HWNetV2 [10]	96.01	-	-
PHOCNet [7]	96.71	-	-
HWNet [9]	98.14	-	-
Proposed Method	**99.6**	**0.5**	**0.5**

Table 4. Results of the Proposed Method on the IAM Dataset and a Comparison of Results to those of other Methods Applied to the Same Dataset

Method	mAP(%)	WER	CER
PHOCNET [7]	72.51	-	-
TPP-PHOCNET [8]	82.74	-	-
HWNet [9]	84.25	5.46	3.00
HWNetV2 [10]	90.65	6.69	3.72
Proposed Method	**82.77**	**17.23**	**8.82**

Table 5. Results, Measured by the Percentage Mean Average Precision (MAP), of the Proposed Method on Various Datasets

METHOD	GW	IFN	IAM
RNN [13]	84.0	-	79.0
HWNET [9]	96.01	-	90.65
PHOCNET [7]	96.71	96.1	72.51
HWNETV2 [10]	98.14	-	89.07
PROPOSED METHOD	**99.6**	**97.9**	**82.77**

5. Conclusion

In this paper, a new approach to word spotting using a convolutional neural network is explained in detail. A novel CNN model with 14 convolutional layers, 6 max pooling and 2 fully connected layers with sigmoid activation is proposed. Four datasets, namely, George Washington, IAM, VML-HD and IFN are used for training and testing the model. It is observed that the discussed architecture outperforms other existing methods in this field. The proposed CNN model outperformed the other method [20] with VML-HD dataset significantly. This result can be considered as an improvement of the current systems in the field of historical Arabic documents because there are not many experiments conducted on such type of documents according to a recent survey paper [23]. Similarly, for other datasets, the mAP is higher when compared to the state-of-the-art methods [7] available in the study. The model has proved to be versatile and robust and can work effectively for multi-language and multi-writers scenarios.

It is noted that the accuracy of the proposed method is higher for datasets with fewer than 1000 classes, while the model cannot accurately analyze datasets with more than 1000 classes. Future studies will include efforts to improve the results on datasets with more than 1000 classes and fine-tuning to improve results on datasets with multiple authors. Further research will explore a real-time implementation of this method for recognition of a word in a live picture and the extension of the method to using transfer learning based on models already trained with the method of this paper.

Acknowledgments

This paper was supported by a QUCP award [QUCPCENG-CSE-15-16-1] from the Qatar University. The statements made herein are solely the responsibility of the authors.

References

[1]. Dey, S., Nicolaou, A., Llados, J., & Pal, U. (2016, November). Local binary pattern for word spotting in handwritten historical document. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)* (pp. 574-583). Springer, Cham.

[2]. Yao, S., Wen, Y., & Lu, Y. (2015, August). Hog based two-directional dynamic time warping for handwritten word spotting. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)* (pp. 161-165). IEEE.

[3]. Mhiri, M., Abuelwafa, S., Desrosiers, C., & Cheriet, M. (2018). Hierarchical representation learning using spherical k-means for segmentation-free word spotting. *Pattern recognition letters*, *101*, 52-59.

[4]. Ghosh, S. K., & Valveny, E. (2015, June). A sliding window framework for word spotting based on word attributes. In *Iberian conference on pattern recognition and image analysis* (pp. 652-661). Springer, Cham.

[5]. Fischer, A., Keller, A., Frinken, V., & Bunke, H. (2010, August). HMM-based word spotting in handwritten documents using subword models. In *2010 20th International Conference on Pattern Recognition* (pp. 3416-3419). IEEE.

[6]. Fischer, A., Frinken, V., Bunke, H., & Suen, C. Y. (2013, August). Improving hmm-based keyword spotting with character language models. In *2013 12th International Conference on Document Analysis and Recognition* (pp. 506-510). IEEE.

[7]. Sudholt, S., & Fink, G. A. (2016, October). Phocnet: A deep convolutional neural network for word spotting in handwritten documents. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)* (pp. 277-282). IEEE.

[8]. Sudholt, S., & Fink, G. A. (2017, November). Evaluating word string embeddings and loss functions for CNN-based word spotting. In *2017 14th iapr international conference on document analysis and recognition (icdar)* (Vol. 1, pp. 493-498). IEEE.

[9]. Krishnan, P., & Jawahar, C. V. (2016, October). Matching handwritten document images. In *European Conference on Computer Vision* (pp. 766-782). Springer, Cham.

[10]. Krishnan, P., & Jawahar, C. V. (2019). Hwnet v2: An efficient word image representation for handwritten documents. *International Journal on Document Analysis and Recognition (IJDAR)*, *22*(4), 387-405.

[11]. Krishnan, P., Dutta, K., & Jawahar, C. V. (2016, October). Deep feature embedding for accurate recognition and retrieval of handwritten text. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)* (pp. 289-294). IEEE.

[12]. Poznanski, A., & Wolf, L. (2016). Cnn-n-gram for handwriting word recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2305-2314).

[13]. Frinken, V., Fischer, A., Manmatha, R., & Bunke, H. (2011). A novel word spotting method based on recurrent neural networks. *IEEE transactions on pattern analysis and machine intelligence*, *34*(2), 211-224.

[14]. Dutta, K., Krishnan, P., Mathew, M., & Jawahar, C. V. (2018, August). Improving CNN-RNN hybrid networks for handwriting recognition. In *2018 16th international conference on frontiers in handwriting recognition (ICFHR)* (pp. 80-85). IEEE.

[15]. Pechwitz, M., & Maergner, V. (2003, August). HMM based approach for handwritten Arabic word recognition using the IFN/ENIT-database. In *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings.* (Vol. 3, pp. 890-890). IEEE Computer Society.

[16]. Azeem, S. A., & Ahmed, H. (2013). Effective technique for the recognition of offline Arabic handwritten words using hidden Markov models. *International Journal on Document Analysis and Recognition (IJDAR)*, *16*(4), 399-412.

[17]. Al Abodi, J., & Li, X. (2014). An effective approach to offline Arabic handwriting recognition. *Computers & Electrical Engineering*, *40*(6), 1883-1901.

[18]. Stahlberg, F., & Vogel, S. (2015, September). The QCRI recognition system for handwritten Arabic. In *International Conference on Image Analysis and Processing* (pp. 276-286). Springer, Cham.

[19]. Rusakov, E., Sudholt, S., Wolf, F., & Fink, G. A. (2018). Expolring architectures for cnn-based word spotting. *arXiv preprint arXiv:1806.10866*.

[20]. Barakat, B. K., Alasam, R., & El-Sana, J. (2018, April). Word spotting using convolutional siamese network. In *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)* (pp. 229-234). IEEE.

[21]. Dreuw, P., Doetsch, P., Plahl, C., & Ney, H. (2011, September). Hierarchical hybrid MLP/HMM or rather MLP features for a discriminatively trained gaussian HMM: a comparison for offline handwriting recognition. In *2011 18th IEEE International Conference on Image Processing* (pp. 3541-3544). IEEE.

[22]. Ahmad, I., Fink, G. A., & Mahmoud, S. A. (2014, September). Improvements in sub-character HMM model based Arabic text recognition. In *2014 14th International Conference on Frontiers in Handwriting Recognition* (pp. 537-542). IEEE.

[23]. Khedher, M. I., Jmila, H., & El-Yacoubi, M. A. (2020). Automatic processing of Historical Arabic Documents: a comprehensive survey. *Pattern Recognition*, *100*, 107144.