DOI: 10.15849/IJASCA.220328.12

Int. J. Advance Soft Compu. Appl, Vol. 14, No. 1, March 2022
Print ISSN: 2710-1274, Online ISSN: 2074-8523

Generative Adversarial Network for an Improved Arabic Handwritten Characters Recognition

Yazan M. Alwaqfi, Mumtazimah Mohamad, and Ahmad T. Al-Taani

Faculty of Informatics and Computing, University of Sultan Zainal Abidin,
Besut Campus, 22200 Besut, Terengganu, Malaysia
e-mail: ywaqfi@yahoo.com
Faculty of Informatics and Computing, University of Sultan Zainal Abidin,
Besut Campus, 22200 Besut, Terengganu, Malaysia
e-mail: mumtaz@unisza.edu.my
Department of Computer Sciences, Faculty of Information Technology and
Computer Sciences, Yarmouk University, Irbid, Jordan
e-mail: ahmadta@yu.edu.jo

Abstract

Currently, Arabic character recognition remains one of the most complicated challenges in image processing and character identification. Many algorithms exist in neural networks, and one of the most interesting algorithms is called generative adversarial networks (GANs), where 2 neural networks fight against one another. A generative adversarial network has been successfully implemented in unsupervised learning and it led to outstanding achievements. Furthermore, this discriminator is used as a classifier in most generative adversarial networks by employing the binary sigmoid cross-entropy loss function. This research proposes employing sigmoid cross-entropy to recognize Arabic handwritten characters using multi-class GANs training algorithms. The proposed approach is evaluated on a dataset of 16800 Arabic handwritten characters. When compared to other approaches, the experimental results indicate that the multi-class GANs approach performed well in terms of recognizing Arabic handwritten characters as it is 99.7% accurate.

Keywords: *Generative Adversarial Networks (GANs), Arabic Characters, Optical Character Recognition, Convolutional Neural Networks (CNNs).*

Received 3 January 2022; Accepted 3 March 2022

1 Introduction

Despite the fact that Optical Character recognition has long been a research focus, automated handwritten characters identification remains a difficult and unsolved issue [1]. Resolving problems in this area is beneficial because it enables the transformation of old manuscripts into digital data, improves convenient storage, and helps find information inside scanned documents.

From a computer's perspective, identifying handwritten manuscripts is considered a complicated issue since individuals has their own style of handwriting. Furthermore, even a single writer's handwriting may differ somewhat from time to time [2]. Since distortions and pattern variations are the most common fundamental challenges in handwriting recognition, feature extraction is critical. In this regard, identifying features manually may result in inadequate data that could lead to inaccurate detection .

The implementation of deep learning in various computer tasks such as supervised learning tasks (segmentation, object detection, and image classification) offered a lot of labelled data to the learning process. This led to significant improvements and enhancements, and it has recently attracted considerable academic interest. Nevertheless, compared to unsupervised learning tasks, like generative models, supervised learning activities have a lower effect via deep learning. Many deep generative models have been proposed, including Deep Boltzmann Machines (DBM) [3], Restricted Boltzmann Machines (RBM) [2], and Variational Autoencoders (VAE) [4]. However, these models have difficulty approximating intractable functions or distributions.

There are various architectures of deep learning including Deep Belief Network, Recurrent Neural Network (RNN), Deep Neural Network (DNN), Convolution Neural Network (CNN), and Generative Adversarial Network (GAN). Using these architectures, deep learning has shown to be a great success in a variety of applications. This article entirely focuses on GAN, a deep neural network type that is commonly used to evaluate visual images.

The generative models have had less effect before the advent of (GANs), which is a robust architecture that is capable of learning very complicated data distributions. According to the game theory, GANs contain two opposing neural networks: a generative model (generator) that generates handwritten digits and characters at random, and a discriminative model (discriminator) which determines whether the generated digits and Characters belong to the real dataset or not. The training of discriminator and generator at the same time is the basic idea of GANs [3].

The generator induces a new data instance or image while the discriminator evaluates the generated instances for identity or originality and decides about whether the generated image belongs to the training set or not. However, the generator attempts to construct another fake image or instance to convince the discriminator that the generated image belongs to the training set. GANs demonstrated and proved their effectiveness in image generation [4], image super-resolution [5], and semi-supervised learning [6]. Even though GANs have outstanding achievements in image processing and generation, it is, however, used in their basic and simple form without any enhancements. As a result, utilizing the discriminator in GANs learning process generally causes instability [7].

The proposed approach consists of three phases. The first is pre-processing of the dataset which aims to remove noise and clean the data. This phase could affect the success of classifying the deep convolutional neural network which will reflect on the results of the OCR accuracy. The pre-processing phase includes two main steps: image resizing and grayscale images.

The second phase is the Generative Adversarial neural networks phase. In this phase, we build and train the GAN networks model (the generator network and the discriminator network) to make it create new reasonable examples from the domain of the problem which is Arabic words and characters. This will make the generator capable of producing real-like images that are almost the same as what is seen in the real world. The new set of generated images will be used later in the deep conventional neural network phase. The third phase is the Deep Conventional neural network phase. This is the final phase of the proposed model, and in it, we build and train the deep conventional neural network on the new dataset of images that were newly generated by both the GAN model and the original dataset. The better the dataset in the training is, the more accurate the model will be. Moreover, achieving the most accurate results relies on the increasing amount of the dataset that is trained on Arabic words .

To avoid over-fitting, we proposed to use Generative and Discriminative algorithms where the training dataset is inputted to the discriminator CNN. Hence, the generator CNN does not have any information about the training dataset, and it generates new data instances as a result. The generator is fed only from the discriminator CNN by the decisions about which of the reviewed instances of data belong to the actual training dataset and which do not. Eventually, this means that implementing GANs certainly helps to avoid over-fitting .Therefore, this paper's major contributions are summarized as follows:

1. It proposes a novel model for Arabic handwritten character detection.
2. The proposed approach combines two different types of deep neural networks: a generative adversarial network, which is called the GAN model, and a deep conventional neural network (DCNN).
3. It evaluates the performance of the proposed model using the AHCD dataset.

2 Related Work

The earliest text generation works relied on Recurrent Neural Networks [8], RNN-like models. However, these works were biased [9] and demonstrated an ill-suited loss function for the sentence generator.

Methods, other than the RNN, were successfully used in various tasks such as classification [10] however, they encountered difficulties from often large data sets in the training process and suffer from different algorithmic and structural limitations[11]. Among these methods are Generative Pre-trained Transformer [12] and Bidirectional Encoder Representations from Transformer [13]. As a result, generative adversarial networks are recommended as both a solution to the challenges and as being more appropriate to the scope of this work.

The problem of Arabic character detection is still being extensively investigated using various techniques and approaches. To overcome this challenge, many approaches are presented. For instance, Deep learning classifiers are used to attain the best recognition accuracy in the Arabic language since deep learning produces reasonably accurate results in the English language.

El-Sawy et al.[1] Performed training of a CNN and then applied it on a collection of 16,800 Arabic handwritten characters from the AHCD. The proposed model used dropout and normalization and two convolutional layers. According to the tested data, the suggested CNN obtained 94.9 % accuracy.

Younis [14] established a CNN to recognize Arabic handwritten characters. Three convolutional layers were proposed, and then they were succeeded by a layer that is totally interconnected within the CNN. Using the AHCD and AIA9K datasets, the CNN obtained a result that is 97.6% accurate based on the AHCD dataset and 94.8% on the AIA9K dataset according to experimental results.

Al-Taani et al. [15] designed a ResNets architecture for Arabic handwritten character apprehension. The proposed approach consisted of three stages: pre-processing phase, training the ResNets on the training set and testing the trained ResNets on the datasets. The results of this approach highlighted that it could achieve accurate rates at 99.8%, 99.05%, and 99.55% using the MADBase, AIA9K, and AHCD, respectively.

AlJarrah et al. [16] presented a CNN model to detect letters and numerals of the written Arabic language. The model was trained using an AHCD dataset. Data augmentation approaches were used in this study to increase model performance and provide improved detection outcomes. The outcomes of the experiment revealed that the given approach could reach a 97.2%success rate. When data augmentation was included, the model gave improved results that were 97.7% accurate.

To enhance Arabic digits detection, Das et al. [17] proposed a collection of 88 characteristics of handwritten numerals' samples in Arabic. 3 layers (input, single hidden, and output) were used to create an MLP classifier. The back-propagation approach was used to train the multi-layer perceptron, which was then utilized to arrange Arabic numerals from the CMATERDB 3.3.1 dataset in classes. The model attained an average accuracy of 94.93% on a database of 3000 samples according to testing results.

Elleuch et al. [18] introduced a Deep Belief Neural Network (DBNN) for the identification of Arabic handwritten characters/words. The proposed model took the row data and proceeded step by step to the unsupervised learning algorithm. This model was tested over the HACDB data set results and achieved 97.9% accuracy.

The study in [19] proposed a Fuzzy classifier with structural characteristics for Arabic handwritten words detection off-line systems based on segmentation processes. The model divided the characters into five categories before extracting the features utilizing the invariant pseudo-Zernike moments. According to the study's findings, the suggested model attained high accuracy at 93.8% for the IFN/ENIT database.

Altwaijry and Al-Turaiki [20] used the Convolutional neural network to develop an identification model for Arabic handwritten characters. They created their own dataset, known as Hijja. To train the model, Hijja and AHCD were utilized, and the Adam optimizer was employed to train the model. The achieved rate of learning was 0.001. The results demonstrated that the model's performance was 97% and 88% accurate according to the AHCD and Hijja datasets in the order given.

In the research in [21], 3 tortuous layers, 2 max-pooling layers, and a layer that is completely interconnected have been proposed for the Arabic handwritten characters model. The suggested model's essential parts were training and recognition. The construction, design, and training of CNN networks with input image data were critical phases in the training phase. The grayscale images of Arabic handwritten characters were cropped and stored. The only pre-processing task used was to convert the image to binary and resize it to 28×28. The results revealed that the suggested model obtained good accuracy of 89.8%, 95.4%, and 92.5% when employing the AHCR, AHCD, and Hijja datasets, respectively.

Alyahya et al. [22] proposed adding a fully connected layer and a Dropout layer to the original ResNet-18 architecture to enhance its ability in recognizing Arabic handwritten characters. The suggested work consisted of four models; two of which utilized a layer that is totally interconnected and either used a dropout layer or did not follow each complicated layer. Also, the other two models used a layer that is totally interconnected and either used a dropout layer or not. They trained and evaluated the CNN-based ResNet18 model using the AHCD dataset; accuracy was 98.3%.

Najadat et al. [23] suggested a model containing 4 convolution layers, 4 Rectified Linear Units (ReLU), 2 max-pooling layers, and 3 fully connected layers. By using the AHCD Dataset, the CNN model achieved 97.2% accuracy.

To identify Arabic Handwritten characters and digits, De Sousa et al. in [24] introduced two DCNN models. The first one was the VGG-12 model which proposed adding a dropout layer before the softmax fully connected layer which served as a classifier. The second model REGU is built from the ground up by incorporating several layers of batch formalization and dropout at CNN layers and FCL. After applying the REGU model on the AHCD database, the results were 99.47% accurate.

Table 1: Summary of deep learning models recognizing handwritten characters

Ref.	Year	Model	Dataset	Accuracy%
[1]	2017	DCNN	AHCD	94.9
[14]	2017	DCNN	AHCD	97.6
[23]	2019	DCNN	AHCD	97.2
[15]	2021	ResNets	AHCD	99.5
[16]	2021	Deep CNN	AHCD	97.7
[17]	2010	MLP	CMATERDB	94.93
[19]	2016	5 Fuzzy ARTMAP	IFN/ENIT	93.8
[18]	2015	DBNN	HACDB	97.9
[20]	2020	DCNN	AHCD	97
[21]	2020	CNN	AHCD	95.4
[24]	2018	DCNN	AHCD	99.4
[22]	2020	CNN	AHCD	98.8

As shown in Table 1, the proposed GAN-based on deep CNN model performed much better than other advanced systems and models when dealing with large datasets available for AHCR in Arabic language text recognition. The obtained results outperformed all the experiments performed in table 1. Because it began with good architecture, the recommended architecture achieved more accuracy.

3 Methodology

Using the most advanced approaches proposed for character recognition and applying them to Arabic character recognition may enhance the efficiency of current approaches. After a critical analysis of the literature in this study, it can be realized that the kind of classifier to be used in the detection process is very important and affects the efficiency of recognition. The primary goal of this work is to evaluate deep learning in terms of its capability to recognize Arabic characters utilizing the semi-supervised learning GANs. The basic architecture of GANs includes a generating network G (z; G) which uses noise as a source to generate data to mislead a discriminator network D(x) that attempts to identify the produced data. By optimizing the generator (G) and the discriminator (S) at the same time, a robust unsupervised originating model, which has the ability to create data that is comparable and closer to the original data images, is learned [25].

The GANs discriminator network in this research is used for more than just determining if a sample is "produced or real." It is also vital for the discriminator network, like a multi-task inside the GANs framework, to be a classifier that separates characteristics from distinct languages [26]. To achieve this, samples from the generator G are added to the actual K classes data set and they were marked with a newly "generated" class $y = K + 1$ to expand the classifier to classes $K + 1$.

The proposed model is broadly divided into three main phases: the pre-processing phase, the Generative Adversarial neural networks phase, and finally the Deep Conventional neural network phase. In addition, the overall design of the proposed model consists of three deep neural networks models; two of them represent the Generative Adversarial neural networks and the last one is the Deep Conventional neural network.

3.1 Preprocessing phase

The pre-processing of the dataset images could affect the success of classifying the deep convolutional neural network which will reflect on the accuracy of the results of the OCR. The pre-processing phase includes two main steps. The first one is image resizing Images are resized to32×32 pixels, so it will be possible to deal with the training dataset images that may vary in size (Height and Width in pixels) at the phases of building deep neural networks layers. As a result, it is necessary to standardize the height and width of these images.

The second step is grayscale images. Images having multi-colored channels pose a challenge for the deep neural network because it faces three types of dissimilar data (R-G-B values of the Red-Green-Blue channels in this case). The deep neural network draws forth the characteristics of those colored images and puts them into their proper classes. This will make the process computationally intensive. The role of the deep conventional neural network is to reduce the images into a form that is easier to process without losing the critical features needed for making a good prediction. Therefore, images are converted into grayscale.

3.2 GANs phase

This phase is concerned with building and training the GAN networks model (the generator network and the discriminator network) to produce new reasonable examples from the problem domain, which are Arabic words and characters. The longer these two neural networks train together, the more sharply they enhance each other's abilities. This results in leading the generator to produce real-like images that are almost the same as what is seen in the real world. The new set of generated images will be used later in the deep conventional neural network phase.

3.2.1 Generator neural network

A neural generator network is trained to maximize the discriminator which predicts that the generated images have a high probability of being real. In addition to that, the generator receives training to reduce the chance that the discriminator is accurate. The cross-entropy loss function is used for this purpose. Cross-entropy can be calculated, where the number of classes N equals 2 in binary classification, as:

$$-(y \log(p)+(1-y) \log(1-p)) \tag{1}$$

For multiclass classification cross-entropy can be calculated as:

$$-\sum c=1Nyo, c\log (po, c) \tag{2}$$

Where N is the number of classes, y is a binary indicator (0 or 1) if category c is the right classification for observation o, p is the indicated possibility, while observation o belongs to category c.

Also, we used the Adaptive Moment Estimation (Adam) as an optimization function for the generator model. As an optimizing algorithm, Adam was presented by [27] to replace the standard random descending procedure to update the weights of an iterative network according to the training data. As a result, Adam is the best among the adaptive optimizers in most cases. Moreover, the adaptive learning rate is Good with sparse data as it is perfect for this type of dataset, straightforward to implement, computationally efficient, and suits large problems as far as data and parameters are concerned.

Finally, for the structure of the generator neural network model, the generator model consists of 10 layers in total. Figure 1 shows the proposed Generator CNN Model layers.

3.2.2 Discriminator neural network

To predict the chance of realness regarding an entered image, which can be demonstrated as a class label of class = x for real and class = 0 for fake, the discriminator neural network model was trained. The discriminator was directly implemented by building its model to predict the chances of 0 for fake images and x for real images and bringing down the loss of cross-entropy where x ∈ [1, 2, ..., n], n = number of real classes. Adam optimization function was used in the discriminator model during the optimization process. The stricter discriminator neural network model consists of 9 layers in total. Figure 2 shows the proposed discriminator CNN Model layers.

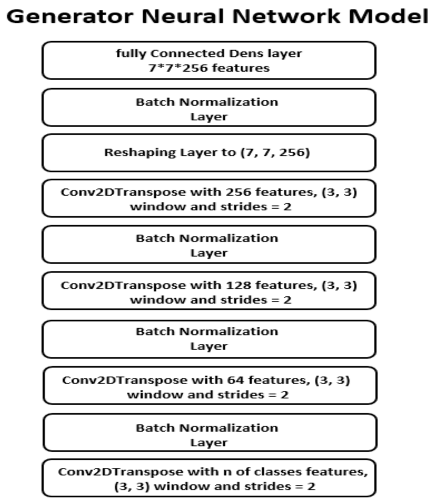

Fig. 1: Proposed generator CNN model layers

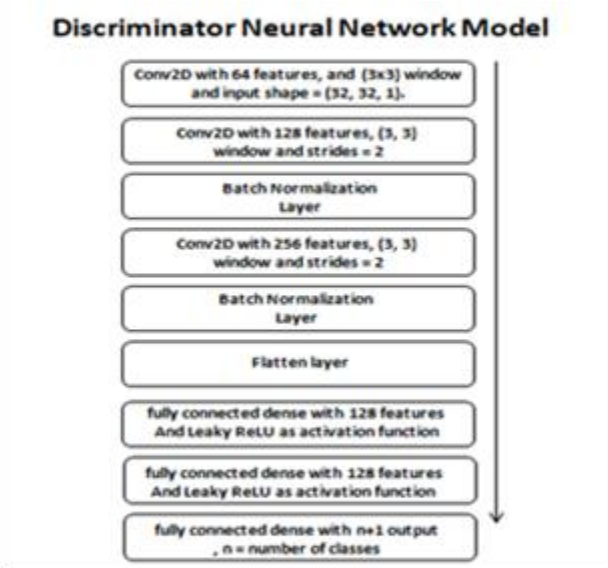

Fig. 2: Discriminator CNN model layer.

3.3 Deep CNN phase

The final phase of the proposed model is to build and train the deep conventional neural network on the new dataset of images, the GAN model's newly generated set of images, and the original dataset. The level of accuracy that the model can achieve depends on how proper and accurate the dataset is in the training results. Moreover, the results will be more accurate if the size of the dataset to be trained on for Arabic words is the largest possible size .

In the case of the proposed approach, Adam optimizer was used for the optimization process in the deep conventional neural network model, while the sparse categorical cross-entropy function was used for the loss function. In addition to that, it is recommended to use sparse categorical cross-entropy when each sample belongs precisely to one class for multiclass classification. Figure 3 shows the overall deep conventional neural network design representing a general framework.

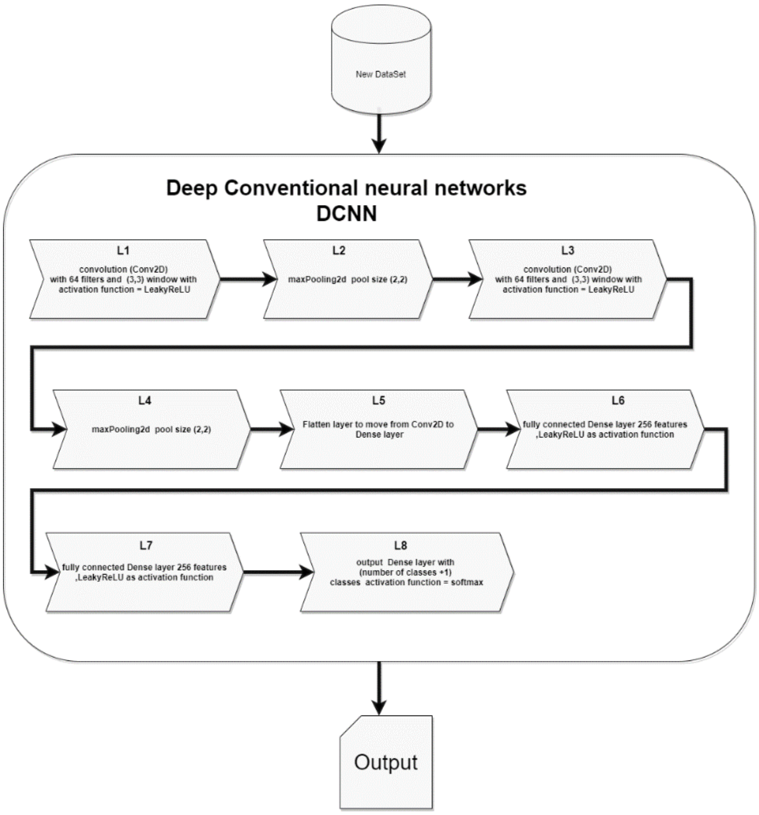

Fig. 3: Deep CNN model

To avoid overfitting, we proposed to use Generative and Discriminative algorithms where the training dataset is inputted to the discriminator CNN. Therefore, the generator CNN does not have any information about the training dataset, hence it creates instances of new data. The generator is fed only from the discriminator CNN by the decisions about which of the reviewed instances of data belong to the actual training dataset and which do not. Figure 4 shows the proposed model for avoiding overfitting in GAN.

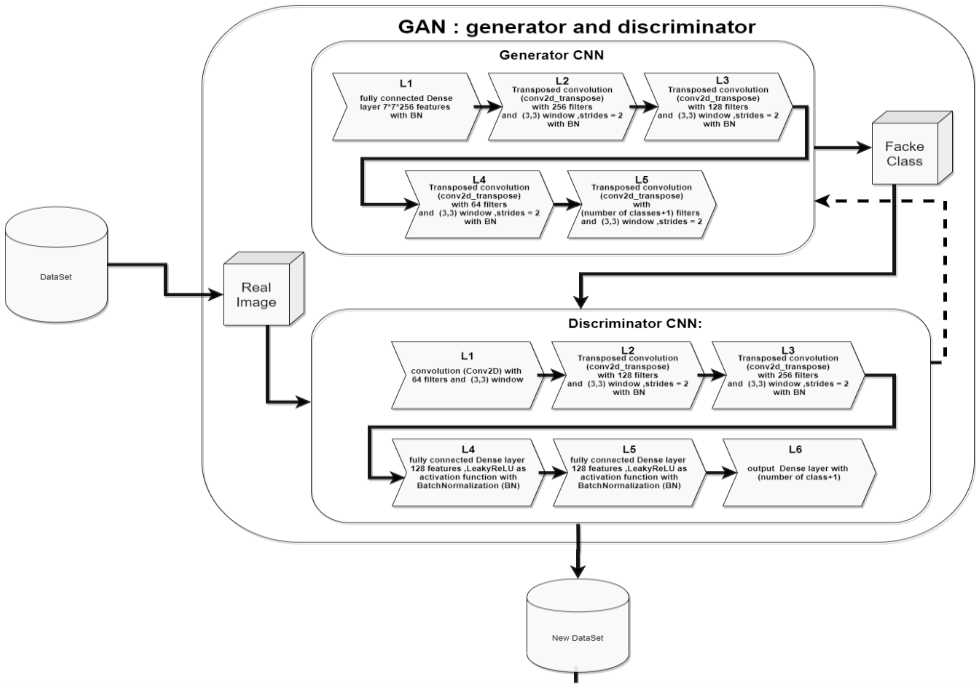

Fig. 4: Proposed model for avoiding overfitting in GAN

4 The Proposed OCR Model

The proposed model architecture consists of three deep neural networks: the GAN neural networks (generator network and the discriminator network) and the traditional deep conventional neural network which is used to train the OCR model. In this work, we used the N+1 class strategy to achieve the proposed goals. This strategy works well for semi-supervised learning when combined with feature matching GANs. The generator creates new images while the discriminator evaluates the generated images for identity or originality. Then, the discriminator decides about whether the generated image belongs to the training set or not. However, the generator tries to generate another fake image or instance to make the discriminator decide that the generated image belongs to the training set. In other words, the generator and the discriminator models compete against each other during the training process.

The generator is trying hard to trick the discriminator while the discriminator is trying hard not to be tricked. This game between the two models motivates both to improve their functionalities as it hones their skills within the lengthy time they spend playing. The discriminator becomes very good at detecting fake images while the generator learns to produce real-like images that are almost the same as what is seen in the real world. We can then ask the GAN for newly created examples from each class (Arabic characters and words).

This will create new instances that presumptively originate from both the existing distribution of samples and the newly generated similar images but are specifically different from the dataset of the existing images. Figure 5 shows the overall research design which represents the general overview of the proposed framework.

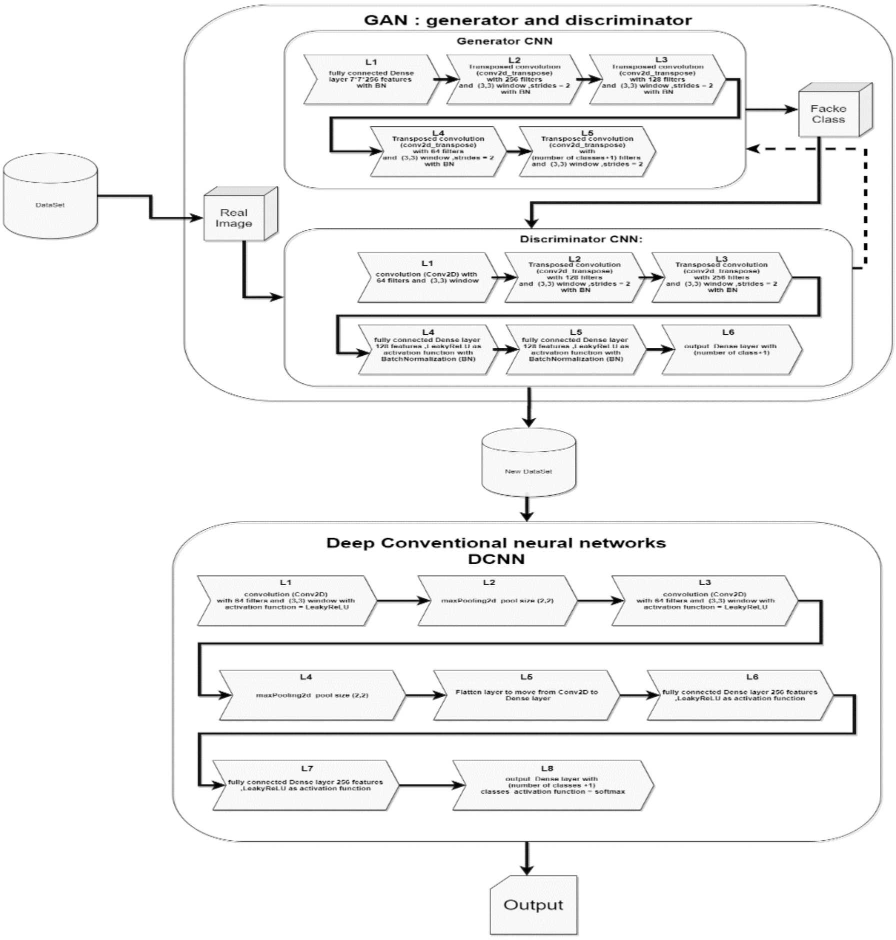

Fig. 5: The proposed OCR model

5 Dataset and Implementation Environment

The proposed model is evaluated with and without the generative adversarial network phase on Arabic Handwritten Characters Dataset (AHCD). This dataset contained 16,800 characters that 60 participants wrote. 90 % of these participants were right-handed and their ages ranged from 19 to 40 years [28] Experiments were carried out using a PC equipped with a Core I5 processor CPU, and 8 GB of RAM, as well as a Windows 10 professional 64 bit, and Python programming environment using the Tensorflow 2.0 and Keras .

6 Experiments Results

The proposed approach was designed from two types of deep neural networks: the first type was generative adversarial networks while the second type was deep conventional neural networks. The proposed approach combined both types of neural networks in a way that generated a new set of images for multiple classes. This new set of images was combined with the original dataset to generate a bigger dataset that enhanced the training accuracy of the proposed OCR model. The experimental results showed that combining the newly generated set of images of Arabic words with the original existing images contributed to making the system produce more accurate results. Also, the increasing dataset size that the system trained on gave more accuracy in the results.

As for image size, the whole training sample images were resized to 32×32 pixels and we repeated the training of the Generator, Discriminator, and the Deep Convolution Neural Networks for 100 and 40 epochs respectively employing a mini-batch size 40 every time. We chose various numbers of epochs at each run incrementally at 10, 20, etc. to attain the highest possible level of accuracy that the proposed approach can offer. We found that, after all those runs were applied, at epoch 40, the best accuracy of the proposed approach topology can be obtained.

In the pre-processing phase, converting the dataset to greyscale images contributed to achieving better testing and validation accuracy in a shorter training time. The complexity caused by dealing with three channeled values was reduced by dealing with one value only, and that conversion process made images easier to process. That was applied to both the GAN model and the CNN model without losing the high accuracy of prediction.

Also, using the Sparse Categorical Cross entropy loss function in the experiments in the three neural networks in the proposed approach solved the problem of needing a long time for conversion. This means that the system will take a longer time to reach the solution which will result in saving memory, time, and computation because it does not use a whole vector, it rather utilizes for a class index only a single integer.

The experiments showed that using activation functions: Leaky ReLU and ReLU did a good job by speeding up the training process. Therefore, the proposed model was faster to learn and converge by a smaller number of epochs. The Leaky ReLU showed more stability than the standard ReLU because it takes zero value when the input values are negative in the training process.

Implementing an additional dropout layer such as Dens, Conv2d, and ConvTranspose layer to prevent overfitting in all the deep neural networks in the proposed model tended to make the model perform worse than it was before adding the dropout layers. The accuracy of the model dropped because of using batch normalization layers between every convolution and activation layer. This is since batch normalization has largely replaced the dropout in modern convolutional architectures because of its regularizing effect. It seems that the dropout layers work better on fully connected layers .

As for activation functions, using Softmax at the last output layer rather than using sigmoid was more suitable for multi-class classification. Sigmoid activation function was not suitable for non-binary classifications since it is equivalent to a 2-element Softmax, where the second element is assumed to be 0. Therefore, sigmoid is mostly used for binary classification.

Finally, implementing sparse categorical cross-entropy loss function for the deep conventional neural network increased the accuracy of the results. This was achieved by overusing the categorical cross-entropy loss function because it is more suitable to the case in which each sample belongs exactly to one class for multiclass classification.

The written code that implemented the proposed model was executed 15 times. The execution accuracy results are presented in Table 2.

Table 2: Experiments results

Exp.	Accuracy	Exp.	Accuracy	Exp.	Accuracy
1	0.97976	**6**	0.98036	**11**	0.98993
2	0.9863	**7**	0.98274	**12**	0.99056
3	0.98333	**8**	0.98363	**13**	0.99468
4	0.98603	**9**	0.98453	**14**	0.99618
5	0.98005	**10**	0.98929	**15**	0.99786

Table 2 clearly shows that the accuracy results for the first five experiments are unsystematic and random. However, after training the model, the results show increasing accuracy after experiment 5, and the highest accurate value is achieved in experiment 15 with an accuracy of 0.99786.

Two experiments are conducted to measure the accuracy of the proposed approach the AHCD dataset, using GANs and without using GANs. Figure 6 shows the results for the two experiments. It can be noticed that the proposed approach achieved best results when GANs is used; it achieved an accuracy of 99.78%, whereas when the GANs is not used the model achieved an accuracy of 96.28%.

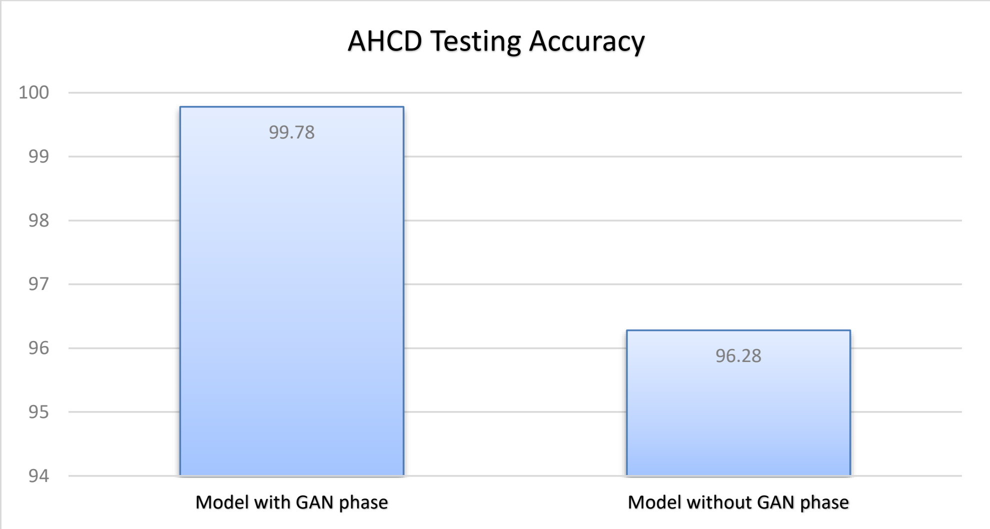

Fig. 6: AHCD testing accuracy for the proposed model with and without GAN

6 Conclusion and Future Work

The study has focused on the improvement of the state-of-art Arabic text recognition by developing a model that optimizes the generative adversarial networks to generate new training images and increase the training dataset. Using this model leads to achieving higher recognition rates. In addition, the GAN-based DCCN model proposed a way to improve a deep neural network model through a new data augmentation technique. This technique adds more images and data to a training set without collecting additional training images. This is a very effective strategy, and it almost always improves the accuracy of recognition. Meanwhile, a technique to handle the overfitting problem is introduced by having more data to train with by using the GAN as Data Augmentation technique. Training with more data can help models avoid overfitting problems. The proposed model using generative adversarial phase achieved competitive accuracy scores where the approach got an average overall test set accuracy of 99.7%.

In future work, the research needs to focus the efforts on adding more enhancements to the accuracy results of text detection. This could be achieved by maximizing the quality of the newly generated images from the generative adversarial networks phase and minimizing the recognition error in the deep conventional neural network phase. In other words, future work needs to focus more on improving the ability of the proposed model regarding text identification.

As a future work for the proposed GAN, based deep conventional neural network model can consider the following:

- The use of more common pre-processing techniques on the dataset images and studies how this affects the success classification and recognition of the deep convolutional neural network and how it will reflect on the results of the OCR accuracy.

- Doing Feature Engineering to extract more information from existing data. As in the hypothesis “Good hypothesis results in good features”, new information is extracted in means of new features. These features may have a higher ability to resolve the variance in the training dataset, thus, resulting in improved model accuracy.
- Uutilizing more data because deep learning algorithms often perform better with more data as mentioned and discussed in the last section. If one cannot reasonably get more data, it is possible to invent more data and create new modified versions of the existing images by using the proposed generative adversarial networks .
- Trying to employ padding, cropping, and scaling techniques of data augmentation in addition to the rotation on the images in the training dataset to increase the training dataset size furthermore.
- Trying various optimizers in the proposed approach to finding the best optimizer during the compilation of the proposed approach. We want to use different optimizers and study the effects of each one on the obtained detection results. There is a need to tune the proposed approach with various optimizers because this can affect the performance of the suggested model.
- Studying the effects of using deferments loss function in the proposed model in both the generative adversarial networks and the deep conventional neural network.
- Using more hidden layers in the proposed approach might slightly improve the accuracy since it depends on the complexity of the problem that needs to be solved. Adding more layers will increase the number of weights in the deep neural networks, and that will increase the complexity of the model. The accuracy of the test data could be reduced if there is not a large training set. This is since an increasingly large network is likely to overfit.

References

[1] El-Sawy, H, Loey, A., & El-Bakry, M. (2017). Arabic handwritten characters recognition using convolutional neural network. WSEAS Trans. Comput. Res. 5(1), 11-19.

[2] Porwal, U., Shi, Z., and Setlur, S. (2013). Machine Learning in Handwritten Arabic Text Recognition. In: (Eds.) Rao, C.R. & Govindaraju, V., Handbook of Statistics, Elsevier, 31, 443-469.

[3] Goodfellow, I. J., Pouget-Abadie, J, Mirza, M, Xu, B, Warde-Farley, D, Ozair, Sh, Courville, A, & Bengio, Y (2014). Generative Adversarial Networks, arXiv:1406.2661v1.

[4] Radford, A, Metz, L, & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.,* pp. 1–16.

[5] Christian, L., Lucas, Th., Huszar, Caballero, F., Jose, Andrew, C., Alejandro, A., Andrew, A., Alykhan, T., Johannes, T., Zehan, W., & Wenzhe, Shi (2017). Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. 105-114. 10.1109/CVPR.2017.19.

[6] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, Xi (2016). Improved techniques for training GANs. *In Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16). Curran Associates Inc., Red Hook, NY, USA, 2234–2242.*

[7] Zhao, J., Mathieu, M., & LeCun, Y. (2017). Energy-based Generative Adversarial Network. arXiv:1609.03126v4**.**

[8] Mikolov, T, Kombrink, S., Burget, L, Černocký, J., & Khudanpur, S. (2011). Extensions of recurrent neural network language model. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5528-5531, doi: 10.1109/ICASSP.2011.5947611.*

[9] Bengio, S., Vinyals, O., Jaitly, N., & Shazeer, N. (2015). Scheduled sampling for sequence prediction with recurrent Neural networks. *In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'15). MIT Press, Cambridge, MA, USA, 1171–1179.*

[10] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. & Sutskever, I. (2018). Language Models are Unsupervised Multitask Learners. http://www.persagen.com/files/misc/radford2019language.pdf. Accessed 8/3/2022.

[11] Zhao, Z., Wallace, E., Feng, S., Klein, D. & Singh, S.. (2021). Calibrate before Use: Improving Few-shot Performance of Language Models. *In Proceedings of Machine Learning Research, 139:12697-12706, Available from https://proceedings.mlr.press/v139/zhao21c.html.*

[12] Budzianowski, P., & VulićHello, I. (2019). It's GPT-2 -- How Can I Help You? Towards the Use of Pretrained Language Models for Task-Oriented Dialogue Systems. arXiv:1907.05774v2**.**

[13] Devlin, J., Ming-Wei C., Kristina, L., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (2018). I*n Proceedings of NAACL-HLT 2019, pages 4171–4186, Minneapolis, Minnesota, Association for Computational Linguistics.*

[14] Younis, K., S. (2018). ARABIC HAND-WRITTEN CHARACTER RECOGNITION BASED ON DEEP CONVOLUTIONAL NEURAL NETWORKS. Jordanian Journal of Computers and Information Technology (JJCIT), 3(3), 186 - 200, doi: 10.5455/jjcit.71-1498142206.

[15] Al-Taani, A., T., & Ahmad, S., T. (2021). RECOGNITION OF ARABIC HANDWRITTEN CHARACTERS USING RESIDUAL NEURAL NETWORKS. Jordanian Journal of Computers and Information Technology (JJCIT), 7(2), 192–205, 2021, doi: 10.5455/jjcit.71-1615204606.

[16] AlJarrah, M. N., Zyout, M. M., & Duwairi, R. (2021). Arabic Handwritten Characters Recognition Using Convolutional Neural Network. *12th International Conference on Information and Communication Systems (ICICS), pp. 182-188, doi: 10.1109/ICICS52457.2021.9464596.*

[17] Das, N., Mollah, A. F., Saha, S., & Haque, S. S. (2010). Handwritten Arabic Numeral Recognition using a Multi Layer Perceptron. *ArXiv, abs/1003.1891*.

[18] Elleuch, M., Tagougui, N., & Kherallah, M. (2015). Arabic handwritten characters recognition using Deep Belief Neural Networks. *IEEE 12th International Multi-Conference on Systems, Signals & Devices (SSD15), pp. 1-5, doi: 10.1109/SSD.2015.7348121.*

[19] Kef, M., Chergui, L., & Chikhi, S. (2016). A novel fuzzy approach for handwritten Arabic character recognition. Pattern Analysis and Applications, 19(4), 1041–1056. https://doi.org/10.1007/s10044-015-0500-4.

[20] Altwaijry, N. & Al-Turaiki, I. (2021). Arabic handwriting recognition system using convolutional neural network. Neural Computing & Applications, 33, 2249–2261. https://doi.org/10.1007/s00521-020-05070-8.

[21] AlKhateeb, J. H., Turani, A., Abuhamdah, A., Sara, M.J., & Klaib, M.F. (2020). An Effective Deep Learning Approach for Improving Off-Line Arabic Handwritten Character Recognition. International Journal Of Software Engineering & Computer Systems (IJSECS), 6(2), 53-61.

[22] Haifa Alyahya1, H., Ben Ismail, M. M., & Al-Salman, A. (2020). Deep ensemble neural networks for recognizing isolated Arabic handwritten characters. ACCENTS Transactions on Image Processing and Computer Vision, 6(21), 68-79. http://dx.doi.org/10.19101/TIPCV.2020.618051.

[23] Najadat, H. M., Alshboul, A. A., & Alabed, A. F. (2019). Arabic Handwritten Characters Recognition using Convolutional Neural Network. *10th International Conference on Information and Communication Systems (ICICS), pp. 147-151.*

[24] Palatnik de Sousa I. (2018). Convolutional ensembles for Arabic Handwritten Character and Digit Recognition. PeerJ Computer Science, 4:e167 https://doi.org/10.7717/peerj-cs.167.

[25] Souly, N., Spampinato, C., & Shah, M. (2017). Semi Supervised Semantic Segmentation Using Generative Adversarial Network. *IEEE International Conference on Computer Vision (ICCV), 2017, pp. 5689-5697, doi: 10.1109/ICCV.2017.606.*

[26] Ding, M., Tang, J., & Zhang, J. (2018). Semi-supervised Learning on Graphs with Generative Adversarial Nets. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*.

[27] Kingma, D. P. & Welling, M. (2014). Auto-encoding Variational Bayes. arXiv:1312.6114v10.

[28] El-Sawy, A., Loey, M., & El-Bakry, H.M. (2017). Arabic Handwritten Characters Recognition Using Convolutional Neural Network. WSEAS Transactions on Computers archive, 5.

Notes on contributors

Yazan M. Alwaqfi is a third-year PhD student of computer science, Faculty of Informatics & Computing at University Sultan Zainal Abidin, Besut, Terengganu, Malaysia. His research interests include artificial intelligence, deep learning, OCR, and technology innovation. Alwaqfi received his MSc in information technology from University Utara Malaysia in 2014. He received his bachelor's degree in Computer Science from Yarmouk University, Jordan in 2011.

Mumtazimah Mohamad completed her BSc in Information Technology (2000) from University Kebangsaan Malaysia, Malaysia, and her MSc. in Software Engineering (2005) from University Putra Malaysia. She received her PhD (2014) from University of Malaysia Terengganu in Computer Science. She has joined University Sultan Zainal Abidin, Terengganu as a lecturer in 2000, she is currently serves as the Deputy Dean for Academic and Postgraduate since 2017. Her research interests and expertise focus on datamining, pattern recognition, machine leaning, neural network and parallel processing.

Ahmad T. Al-Taani is a full professor of AI and a researcher at the department of computer sciences, faculty of information technology and computer sciences at Yarmouk University, Jordan. He obtained his PhD, MSc, and BSc degrees from university of Dundee (UK), National University (USA), and Yarmouk University (Jordan) in 1994, 1988 and 1985 respectively. Dr. Al-Taani has served on the editorial board for journal of pattern recognition research and a reviewer for several reputed journals including natural language engineering, soft computing, applied soft computing, cognitive computation, cluster computing, and others. His expertise areas include NLP, Machine Learning, Knowledge-based Systems, Image Processing, Pattern Recognition, and Blockchain Technologies.