# CNN for Handwritten Arabic Digits Recognition Based on LeNet-5

Ahmed El-Sawy[1], Hazem EL-Bakry[2], and Mohamed Loey[1(✉)]

[1] Faculty of Computer and Informatics, Computer Science Department,
Benha University, Benha, Egypt
{ahmed.el_sawy,mohamed.loey}@fci.bu.edu.eg
[2] Faculty of Computer and Information Sciences, Information System Department,
Mansoura University, Mansoura, Egypt
helbakry5@yahoo.com

**Abstract.** In recent years, handwritten digits recognition has been an important area due to its applications in several fields. This work is focusing on the recognition part of handwritten Arabic digits recognition that face several challenges, including the unlimited variation in human handwriting and the large public databases. The paper provided a deep learning technique that can be effectively apply to recognizing Arabic handwritten digits. LeNet-5, a Convolutional Neural Network (CNN) trained and tested MADBase database (Arabic handwritten digits images) that contain 60000 training and 10000 testing images. A comparison is held amongst the results, and it is shown by the end that the use of CNN was leaded to significant improvements across different machine-learning classification algorithms.

## 1 Introduction

Recognition is an area that covers various fields such as, face recognition, image recognition, finger print recognition, character recognition, numerals recognition, etc. [1]. Handwritten Digit Recognition system (HDR) is an intelligent system able to recognize handwritten digits as human see. Handwritten digit recognition is an important component in many applications; check verification, office automation, business, postal address reading and printed postal codes and data entry applications are few examples [2]. The recognition of handwritten digits is a more difficult task due to the different handwriting styles of the writers.

Over the last few years, deep learning [3] are the most researched area in machine learning that model hierarchical abstractions in input data with the help of multiple layers. Deep learning techniques have achieved state-of-the-art performance in computer vision [4,5], big data [6,7], automatic speech recognition [8,9] and in natural language processing [10]. Although, increase in computing power has contributed significantly to the development of deep learning techniques, deep learning techniques attempts to make better representations and create models to learn these representations from large-scale data.

Deep learning have many architectures such as Convolution Neural Networks (CNN). CNN is a multi-layer feed-forward neural network that extract properties from the input data. CNN trained with neural network back-propagation algorithm. CNN have the ability to learn complex, high-dimensional, non-linear mappings from very large number of data (images). Moreover, CNN shows an excellent recognition rates for characters and digits recognition [11]. The advantage of CNN is that it automatically extracts the salient features which are invariant and a certain degree to shift and shape distortions of the input characters [12].

Feature extraction is an important key factor to make a successful recognition system. The recognition system requires that features distinguished characteristics among different labels while retaining invariant characteristics within the same labels. Traditional feature extraction that designed by hand is really a boring and time consuming task and cannot process raw images, but the automatic extraction techniques can restore and reconstruct features directly from raw images. Based on CNN [11] extract feature from trainable dataset can done automatically. So, the propose of the paper is using CNN to create deep learning recognition system for Arabic handwritten digits recognition.

The rest of the paper is organized as follows: Sect. 2 gives a review on some of the related work done in the area. Section 3 describes the motivation and proposed approach, Sect. 4 gives an overview of the dataset and results, and we list our conclusions and future work in Sect. 5.

## 2   Related Work

Various methods have been proposed and high recognition rates are reported for the recognition of English handwritten digits [13–15]. Niu and Suen [13] proposed recognize handwritten digits using Convolutional Neural Network (CNN) and Support Vector Machine (SVM). There Experiments have been conducted on MNIST digit database. They achieve recognition rate of 94.40 % with 5.60 % with rejection. Tissera and McDonnell [14] introduced a supervised auto-encoder architecture based on extreme machine learning to classify Latin handwritten digits based on MNIST dataset. The proposed technique can correctly classify up to 99.19 %. Ali and Ghani introduced Discrete Cosine Transform based on Hidden Markov models (HMM) to classify handwritten digits. They used MNIST as training and testing datasets. HMM have been applied as classifier to classify handwritten digits dataset. The algorithm provides promising recognition results on average 97.2 %.

In recent years many researchers addressed the recognition of text including Arabic. In 2011, Melhaoui et al. [16] proposed an improved method for recognizing Arabic digits based on Loci characteristic. Their work is based on handwritten and printed numeral recognition. The recognition is carried out with multi-layer perceptron technique and K-nearest neighbour. They trained there algorithm on dataset contain 600 Arabic digits with 200 testing images and 400 training images. They were able to achieve 99 % recognition rate on small database.

In 2008, Mahmoud [17] proposed a technique for the automatic recognition of Arabic handwritten digits using Gabor-based features and Support Vector Machines (SVMs). They used a medium database have 21120 samples written by 44 writers. The dataset contain 30 % for testing and the remaining 70 % of the data is used for training. They achieved average recognition rates are 99.85 % and 97.94 % using 3 scales & 5 orientations and using 4 scales & 6 orientations, respectively.

In 2014, Takruri et al. [18] presented three level classifier based on Support Vector Machine, Fuzzy C Means and Unique Pixels for the classification of handwritten Arabic digits. They tested the new algorithm on a public dataset. The dataset contain 3510 images with 40 % are used for testing and 60 % of images are used for training. The overall testing accuracy reported is 88 %.

In 2013, Pandi Selvi and Meyyappan [1] presented a method to recognize Arabic digits using back propagation neural network. The final result shows that the proposed method provides an recognition accuracy of more than 96 % for a small sample handwritten database.

In 2014, Majdi Salameh [19] proposed two methods about enhancing recognition rate for typewritten Arabic digits. First method that calculates number of ends of the given shape and conjunction nodes. The second method is fuzzy logic for pattern recognition that studies each shape from the shape, and then classifies it into the numbers categories. Their proposed techniques was implemented and tested on some fonts. The experimental results made high recognition rate over 95 %.

In 2014, AlKhateeb et al. [20] presented a system to classify Arabic handwritten digit recognition using Dynamic Bayesian Network. They used discrete cosine transform coefficients based features for classification. Their system trained and tested on Arabic digits database (ADBase) [21] which contains 70,000 Arabic digits. They reported average recognition accuracy of 85.26 % on 10,000 testing samples.

## 3   Proposed Approach

### 3.1   Motivation

Arabic digits recognition and different handwriting styles as well, making it important to find and work on a new and advanced solution for handwriting recognition. A deep learning systems needs a huge number of data to be able to make good decisions. In [1,16–18] they applied the algorithms on a small handwritten images, the problem is the small database of training and testing images. In [21] proposed a large Arabic handwriting digits database called (MADBase) with training and testing images. The proposed database of images gives us with large different handwriting styles. So, the use of MADBase database and deep learning lead to the suggestion of our approach.

### 3.2 Suggested Approach

Convolutional neural networks (CNN) are a class of deep models that were inspired by information processing in the human brain. In the visual of the brain, each neuron has a receptive field capturing data from certain local neighborhood in visual space. They are specifically designed to recognize multi-dimensional data with a high degree of in-variance to shift scaling and distortion.
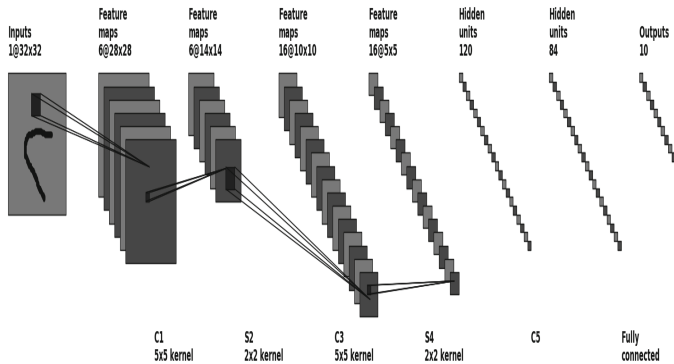


**Fig. 1.** Convolutional neural networks LeNet-5

CNN architecture is made of one input layer and multi-types of hidden layers and one output layer. The fist kind of hidden layers is responsible for convolution and the other one is responsible for local averaging, sub sampling and resolution reduction. The third hidden layers act as a traditional multi-layer perceptron classifier.

In this study LeNet-5 CCN architecture is used with an 8 layers including one input layer, one output layer, two convolutional layers and two sub-sampling for automatic feature extraction, two fully connected layers as multi-layer percep-tron hidden layers for nonlinear classification. The CNN architecture is shown in Fig. 1. The input image size is $32 \times 32$ with 1 channel (i.e. grayscale image).

The first convolutional layer C1 is a convolution layer with 6 feature maps and a $5 \times 5$ kernel for each feature map. There are inputs for each neuron of C1 planes, which is obtained from a $5 \times 5$ receptive field at the previous (input) layer. According to weights sharing strategy, all units in these feature maps use the same weights and bias to produce a linear location invariant filter to be applied to all regions of the input image. The sharing weights of this layer will be adapted during training procedure. This layer C1 has six different biases and six different $5 \times 5$ kernels including 156 trainable parameters, 4704 number of neurons and 122304 connections. The next layer is a sub sampling Layer S2 with six feature maps and a $2 \times 2$ kernel for each feature map.

In fact, after averaging the input samples in the receptive field of an out-put pixel, the result is multiplied and added by two trainable coefficients which

**Table 1.** CNN layers description for our approach

| Lenet5 layers | Description | |
|---|---|---|
| Layer 1 [Input] | number of feature maps:1<br>number of neurons:0<br>number of connections:0 | number of parameters:0<br>number of trainable parameters:0 |
| Layer 2 [C1] | number of feature maps: 6<br>number of neurons:4704<br>number of connections:122304 | number of parameters:156<br>number of trainable parameters:156 |
| Layer 3 [S2] | number of feature maps:6<br>number of neurons:1176<br>number of connections:5880 | number of parameters:12<br>number of trainable parameters:12 |
| Layer 4 [C3] | number of feature maps:16<br>number of neurons:1600<br>number of connections:151600 | number of parameters:1516<br>number of trainable parameters:1516 |
| Layer 5 [S4] | number of feature maps:16<br>number of neurons:400<br>number of connections:2000 | number of parameters:32<br>number of trainable parameters:32 |
| Layer 6 [C5] | number of feature maps:120<br>number of neurons:120<br>number of connections:48120 | number of parameters:48120<br>number of trainable parameters:48120 |
| Layer 7 [F6] | number of feature maps:10<br>number of neurons:10<br>number of connections:1210 | number of parameters:1210<br>number of trainable parameters:1210 |

are assumed to be similar for the output pixels of a feature map, but different in different feature maps. This layer has 12 trainable parameters and 5880 connections.

The third layer C3 is a convolution layer with 16 feature maps and a $5 \times 5$ kernel for each feature map acts as previous convolutional layer. This layer has 16 feature maps and each neuron of each output feature map connects to some $5 \times 5$ pixels areas at the previous layer S2. The next layer S4 is a sub-sampling layer with 16 feature maps and a $2 \times 2$ kernel for each feature map. Layer S4 has 32 trainable parameters and 2000 connections. Layer C5 is a convolution layer with 120 feature maps and a $6 \times 6$ kernel for each feature map which has 48120 connections and trainable parameters. Layer F6 is the last fully connected layer which selected 84 neurons. The final layer is the output layer that has 10 neurons for 10 digit classes. The output of each CNN layers for Arabic digit 4 illustrated in Fig. 2. Each CNN layers have number of feature maps, neurons, connections, parameters, trainable parameters. All this parameters describe in Table 1.
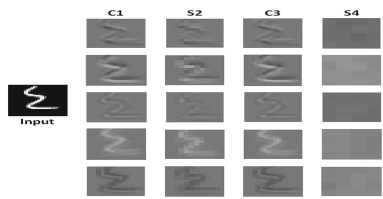
**Fig. 2.** Layers output for digit "4"

## 4   Experiment

### 4.1   Dataset

El-sherif and Abdleazeem released an Arabic handwritten digit database (ADBase) and modified version called (MADBase) [21]. The MADBase is a modified version of the ADBase benchmark that has the same format as MNIST benchmark [22]. ADBase and MADBase are composed of 70,000 digits written by 700 writers.
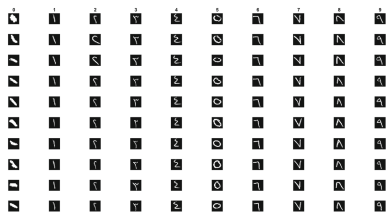


**Fig. 3.** Sample of MADBase benchmark training database

Each writer wrote each digit (from $0-9$) ten times. To ensure including different writing styles, the database was gathered from different institutions: Colleges of Engineering and Law, School of Medicine, the Open University (whose students span a wide range of ages), a high school, and a governmental institution. The databases is partitioned into two sets: a training set (60,000 digits to 6,000
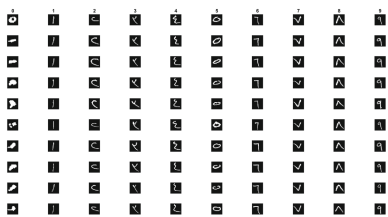


**Fig. 4.** Sample of MADBase benchmark testing database

images per class) and a test set (10,000 digits to 1,000 images per class). The ADBase and MADBase is available for free (http://datacenter.aucegypt.edu/shazeem/) for researchers. Figures 3 and 4 shows samples of training and testing images of MADBase database.

## 4.2   Results

In this section, the performance of CNN was investigated for training and recognizing Arabic characters. For the setting architecture, a convolutional layer is parameterized by the size and the number of the maps, kernel sizes, skipping factors. This section describes about our attempt to apply the CNN based Arabic digits classification. The experiments are conducted in MATLAB 2016a programming environment. LeNet-5 network is used for implementing CNN on Arabic digits on MADBase database. At first for evaluating the performance of CNN on Arabic digits, incremental training approach was used on the proposed approach. We started with 4 classes for training first and calculated the accuracy. Then number of classes is slowly increased. As shown in Fig. 5, the Root Mean Square Error (RMSE) of the proposed approach is .894 for training data and 1.105 fot testing data. The miss-classification rate is come down to 1 % for training and 12 % for testing shown in Fig. 6. Here the algorithm is trained for 30 iterations, but from iteration 12 itself the network shows a good accuracy.

In Fig. 7, confusion matrix illustrated that the first two diagonal cells show the number and percentage of correct classifications by the trained network. For
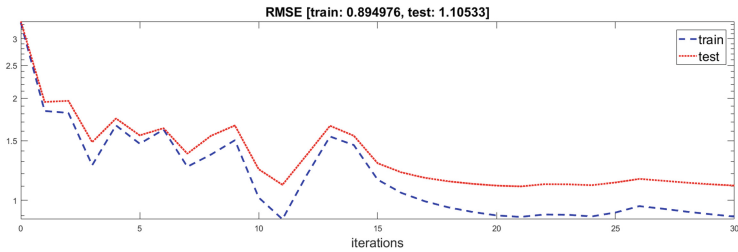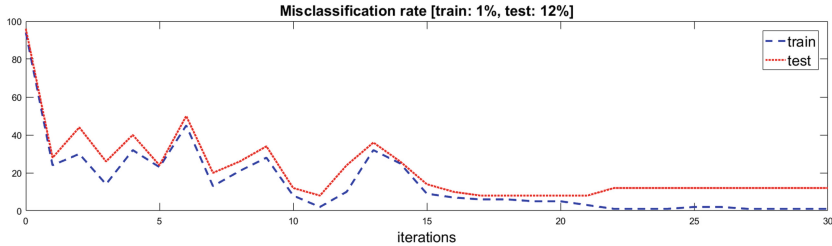


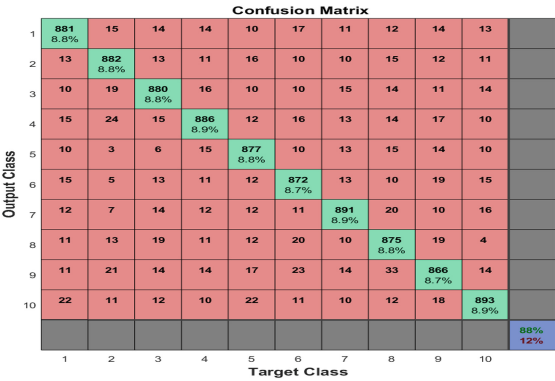**Fig. 5.** Root Mean Square Error



**Fig. 6.** Miss-classification rate

**Fig. 7.** Confusion matrix

**Table 2.** Comparison between proposed approach and other approach

| Authors | Database | Training data<br>Testing Data | Misclassification<br>error |
|---------|----------|-------------------------------|----------------------------|
| Takruri et al. [18] | Public database | 3510 images<br>60 % training digits<br>40 % testing digits | 12 % |
| AlKhateeb et al. [20] | ADBase | 60,000 training digits<br>10,000 testing digits | 14.74 % |
| Majdi Salameh [21] | Fonts | 1000 training digits<br>1000 testing digits | 5 % |
| Melhaoui et al. [16] | Private database | 600 images<br>400 training digits<br>200 testing digits | 1 % |
| Pandi Selvi and Meyyappan [1] | Private database | Sample handwritten images are tested | 4 % |
| Mahmoud [17] | Private database | 21120 images<br>70 % training digits<br>30 % testing digits | 0.15 % and<br>2.16 % |
| Our approach | MADBase | 70000 images<br>60000 training digits<br>10000 testing digits | **1 % training**<br>**12 % testing** |

example 881 of class (1) are correctly classified as class (1). This corresponds to 8.81 % of all 1000 test images of class (1). The column in the target class show the miss-classification of the class. Overall, 88 % of the predictions are correct and 12 % are wrong classifications.

Finally, in Table 2 shown the obtained results with CNN on MADBase database. It can be seen from Table 2 that the proposed approach have the large database and have the best miss-classification error. The results are better than

the results reported in related work [1,16–18,20], although it is sometimes hard to compare, because previous work has not experimented with large database benchmark. The proposed method obtained 1 % miss-classification error on training data and 12 % miss-classification error on testing data.

## 5    Conclusion and Future Work

Handwritten Character Recognition for Arabic digits is an active research area which always needs an improvement in accuracy. This work is based on recognition of Arabic digits using convolution neural networks (CNN). The approach were tested on a large Arabic digits database (MADBase). As experimental results, the approach gives best accuracy in large database with 1 % training miss classification error rate and 12 % testing miss classification error rate. Our future work will be focusing on improving the performance of handwriting Arabic digits recognition using other improved deep learning techniques.

## References

1. Selvi, P.P., Meyyappan, T.: Recognition of Arabic numerals with grouping and ungrouping using back propagation neural network. In: International Conference on Proceedings of Pattern Recognition, Informatics and Mobile Engineering (PRIME), pp. 322–327 (2013)
2. Mahmoud, S.: Recognition of writer-independent off-line handwritten Arabic (Indian) numerals using hidden Markov models. Sig. Process. **88**(4), 844–857 (2008)
3. Lecun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
4. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 25, pp. 1097–1105. Curran Associates Inc., Red Hook (2012)
5. Afaq Ali Shah, S., Bennamoun, M., Boussaid, F.: Iterative deep learning for image set based face and object recognition. Neurocomputing **174**, 866–874 (2016)
6. Zhang, Q., Yang, L.T., Chen, Z.: Deep computation model for unsupervised feature learning on big data. IEEE Trans. Serv. Comput. **9**(1), 161–171 (2016)
7. Chen, X.W., Lin, X.: Big data deep learning: challenges and perspectives. IEEE Access **2**, 514–525 (2014)
8. Cai, M., Liu, J.: Maxout neurons for deep convolutional and LSTM neural networks in speech recognition. Speech Commun. **77**, 53–64 (2016)
9. Sainath, T.N., Kingsbury, B., Saon, G., Soltau, H., Mohamed, A.-R., Dahl, G., Ramabhadran, B.: Deep convolutional neural networks for large-scale speech tasks. Neural Netw. **64**, 39–48 (2015)
10. Collobert, R., Weston, J., Bottou, O., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**, 2493–2537 (2011)

11. Maitra, D.S., Bhattacharya, U., Parui, S.K.: CNN based common approach to handwritten character recognition of multiple scripts. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 1021–1025 (2015)
12. Yu, N., Jiao, P., Zheng, Y.: Handwritten digits recognition base on improved LeNet5. In: Proceedings of Control and Decision Conference (CCDC), 2015 27th Chinese, pp. 4871–4875 (2015)
13. Niu, X.-X., Suen, C.Y.: A novel hybrid CNNSVM classifier for recognizing handwritten digits. Pattern Recogn. **45**(4), 1318–1325 (2012)
14. Tissera, M.D., McDonnell, M.D.: Deep extreme learning machines: supervised autoencoding architecture for classification. Neurocomputing **174**, 42–49 (2016)
15. Ali, S.S., Ghani, M.U.: Handwritten digit recognition using DCT and HMMs. In: 2014 12th International Conference on Frontiers of Information Technology (FIT), pp. 303–306 (2014)
16. Melhaoui, O.E., Hitmy, M.E., Lekhal, F.: Arabic numerals recognition based on an improved version of the loci characteristic. Int. J. Comput. Appl. **24**(1), 36–41 (2011)
17. Mahmoud, S.A.: Arabic (Indian) handwritten digits recognition using Gabor-based features. In: International Conference on Proceedings of Innovations in Information Technology, IIT 2008, pp. 683–687 (2008)
18. Takruri, M., Al-Hmouz, R., Al-Hmouz, A.: A three-level classifier: fuzzy C means, support vector machine and unique pixels for Arabic handwritten digits. In: World Symposium on Proceedings of Computer Applications & Research (WSCAR), pp. 1–5 (2014)
19. Salameh, M.: Arabic digits recognition using statistical analysis for end/conjunction points and fuzzy logic for pattern recognition techniques. World Comput. Sci. Inf. Technol. J. **4**(4), 50–56 (2014)
20. Alkhateeb, J.H., Alseid, M.: DBN - based learning for Arabic handwritten digit recognition using DCT features. In: 2014 6th International Conference on Computer Science and Information Technology (CSIT), pp. 222–226 (2014)
21. Hafiz, A.M., Bhat, G.M.: Boosting OCR for some important mutations. In: Second International Conference on Advances in Computing and Communication Engineering (ICACCE), pp. 128–132 (2015)
22. Wu, H., Gu, X.: Towards dropout training for convolutional neural networks. Neural Netw. **71**, 1–10 (2015)