

Arabic handwritten digits recognition based on convolutional neural networks with resnet-34 model

Rasool Hasan Finjan, Ali Salim Rasheed, Ahmed Abdulsahib Hashim, Mustafa Murtdha

Department of Media Technology Engineering, College of Engineering, University of Information Technology and Communications, Baghdad, Iraq

Article Info

Article history:

Received Feb 25, 2020

Revised Jul 4, 2020

Accepted Aug 4, 2020

Keywords:

Convolutional neural networks

Deep learning

Handwritten digits recognition

Resnet model

Transfer learning

ABSTRACT

Handwritten digits recognition has attracted the attention of researchers in pattern recognition fields, due to its importance in many applications in public real life, such as read bank checks and formal documents which is a continuous challenge in the last years. For this motivation, the researchers created several algorithms in recognition of different human languages, but the problem of the Arabic language is still widespread. Concerning its importance in many Arab and Islamic countries, because the people of these countries speak this language, However, there is still a little work to recognize patterns of letters and digits. In this paper, a new method is proposed that used pre-trained convolutional neural networks with resnet-34 model what is known as transfer learning for recognizing digits in the arabic language that provides us a high accuracy when this type of network is applied. This work uses a famous arabic handwritten digits dataset that called MADBase that contains 60000 training and 1000 testing samples that in later steps was converted to grayscale samples for convenient handling during the training process. This proposed method recorded the highest accuracy compared to previous methods, which is 99.6%.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Rasool Hasan Finjan

University of Information Technology and Communications

College of Engineering, Baghdad, Iraq

Email: rasool.hasan@uoitc.edu.iq

1. INTRODUCTION

Recognition is an important area of machine learning that has contributed to the areas of facial recognition, image recognition, character recognition, digit recognition, etc. handwritten digit recognition [1] has been receiving a perceptible interest from researchers over the past years. It is the active tool that used in commercial importance in fields such as check reading, collect data from forms, and textbook digitization. Therefore, this field has become an effective element in many applications in public life. Arabic is the sixth official language, adopted by the United Nations [2], it is one of the languages widely used all over the world and there are more than 290 million Arabic speakers across the world. Since the greater part of the past work concentrated about depending on Latin languages, However, the Arabic language did not receive much interest in research in this field, that leads to creating more challenge for us. One of the most important things that we note in the Arabic language, the words are written from right to left while the digits are written from left to right, Figure 1 shows ten numbers of classes in the Arabic form.

To address a problem of recognizing Arabic digits, many techniques have been used for this task and the most effective one is deep learning [3], deep learning methods have outperformed state-of-the-art alternatives in speech recognition, object detection, face recognition and Especially in the field of identifying

different international languages, it has proven its effectiveness in various languages, including Romanian [4], the working standards of deep learning method is how to extract the features from large scale datasets starting from the first layer and transform the representation through hidden layers and little by little these transformations become a more abstract and more complex model.

English Digits	0	1	2	3	4	5	6	7	8	9
Arabic Digits	٠	١	٢	٣	٤	٥	٦	٧	٨	٩

Figure 1. Ten numbers digits in the Arabic language

Presently, one of the most noteworthy methods of deep learning algorithms is convolutional neural networks (CNNs) [5, 6]. CNNs are similar to artificial neural networks (ANNs) they consist of neurons that receive an input from the previous layer and computes the output by non-linear function such as the sigmoid function [7], but the enormous difference is that CNNs are used in the computer vision fields such as image classification [8], for the reason that it needs fewer parameters during the training the model and therefore its Makes less time consumption.

1.1. Related works

The method proposed in [9] to recognize Arabic numerals uses a multi-layer perceptron (MLP) as a classifier, where a set of 88 features is used. The feature set includes 72 shadow features and 16 octant features. This proposed method uses a dataset as a CMATERDB 3.3.1 that contains a 3000 handwritten sample, all of these samples are scaled to the size of 32x32 pixels. For regular training step image is converted in grayscale form, the architecture of this algorithm is composed of a single input layer, a single hidden layer, and a single output layer [10]. The motivation to utilize a single hidden layer because it is sufficient to classify the given dataset. For the non-linearity function, the author used a sigmoid function after each layer and uses a back-propagation algorithm [11] for training the classifier. The neurons number was changed in accordance with 54 that records an accuracy of 93.8% in recognition Arabic numerals.

The method introduced in [12] suggested an improved method that used the same dataset as [6], but this time the CNNs used for digit recognizing problems, the images are normalized to become an accepted form for CNNs conditions. The model used in this adopted method consists of two convolution layers with kernels size of 5x5 and 3x 3 respectively, next the pooling layer presented that takes the maximum value with a pool size of 2x 2. The fully connected layer consists of 128 neurons and the final layer that represents output contains 10 neurons for each digit class. ReLU function [13] is used as an activation function and categorical cross-entropy [14] to determine the error accrues through the back-propagation stage, and finally, the Softmax function [15] is used to get the conclusive outcome from the output layer. The accuracy achieved by this model is 97.4% over the validation dataset.

Another work is introduced in this field for recognizing Arabic digits [16], two changes are made on the method that described in [6] to improve the accuracy, the data augmentation strategy [17] which is increasing the number of images was added to the given dataset to solve the problem of overfitting. And then the activation function is changed from ReLU to ELU [18] to provide more robustness to the vanishing gradient problem. The model of this method is composed of four convolution layers, each layer followed by ELU as an activation function with the kernels of size 3x3. a pooling layer (max-pool) was added with size 2x2. Next step, the processed images are flattened and will be the input for the fully connected layers, and finally, the last layer (output) contains 10 neurons that represent 10 classes for classifying the 10 digits. To determine the final result, the softmax function was used to compute the probability for each class that represents a digit [8]. For reducing the effect of overfitting, the authors suggest using a dropout rate 25% for each convolution and fully connected layers, it means that after one epoch 25% of neurons are dropped and make their weights zero to make it ineffective in the network. This proposed method gave accuracy 99.4%, which is considered a better result than previous works.

2. THE PROPOSED METHOD

To deal with Hand Written digits recognition problems, the model can be built from scratch and starting the recognition operation, but in this way, we shall get into trouble and are consuming more time during the training process, especially if the dataset is very large and the training process may take days and

perhaps weeks with high computing capabilities. The modern approach nowadays it uses a remarkable technique called transfer learning [19, 20] which is a popular method in computer vision and NLP fields because it enables us to build a high-accuracy model with little training time. There are various strategies in working with transfer learning and one of these strategies is fine-tuning [21, 22] which is means taking weights of the pre-trained model and use it as initialization for a new model that we want to train on data from the same domain for example images, it used to speed up the training process and overcome small dataset size problems.

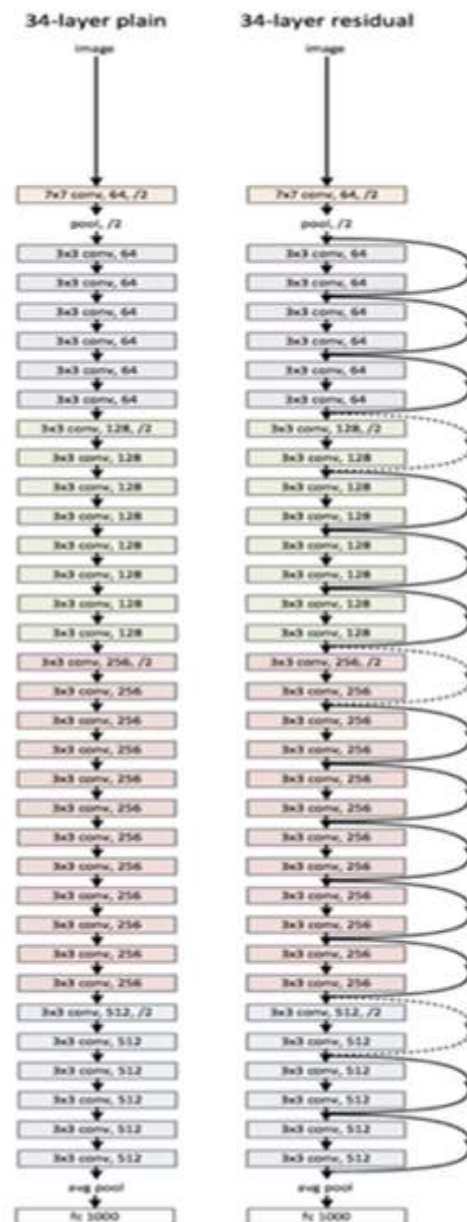


Figure 2. (Right) a residual network with 34 layers, (left) a plain network with 34 layers from the original paper [23]

Consequently, most layers from a pre-trained model are useful in configuring a new model because most computer vision problems involve similar low-level visual patterns. The proposed method in this paper to training our model is the ResNets model [23]. One of the problems ResNets solve is the vanishing gradient. This problem occurs when the network is very deep, the gradients where the loss function is calculated go directly to zero. No

learning is executed because the weights not updating its values. By using the ResNets model, the gradients can across directly through the shortcut connections to the rear from previous layers to initial filters, so will reuse most of the pre-trained Resnet model layers and just replace the final layer that use to make predictions.

As shown in Figure 2, on the right side, we can see the ResNets model based on the plain network, which is shown in the left of the figure. Each colored block of layers represents a series of convolutions of a similar dimension. All of the layers have the same pattern, they consist of 3x3 convolution with fixed feature map dimensions 64, 128, 256, 512 respectively, the dimensions that presented as a solid line remain the same size when the input flow across every 2 convolution layer. The dotted lines which present the change in dimensions of the input volume. We noticed that reduction in their dimension due to convolution operation, this reduction between layers occurs by the size of stride from 1 to 2, on the first convolution for each layer. Every layer of a ResNet is composed of several blocks, this is because when ResNet go deeper, they regularly do it by expanding the number of tasks inside a block, but the number of the total layers continues as before.

The core idea behind the shortcut connections is the way that our network has skip layers that illustrated in Figure 3, therefore, for instance, the two convolution layers it takes as input bunch of feature maps x and the output it will present as $H(x)$. To perform the output of this block, the input x is copied to output $H(x)$, so we have to skip two layers in the network, so the output becomes $H(x) = F(x) + x$. with this concept, instead of learning $H(x)$ we would learn the residual $F(x)$ its equation will become $F(x) = H(x) - x$, therefore in the worst scenario, if the weights are not updated, every value becomes a small or zero, therefore at least the x will be learned.

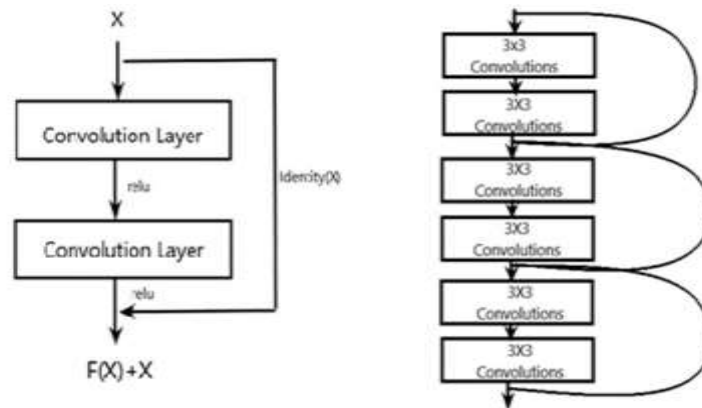


Figure 3. The shortcut connections in the residual network

3. EXPERIMENT AND RESULTS

As mentioned before, The MADBase Arabic handwritten digit dataset is used in our method to recognize Arabic digits, It contains 60000 training and 1000 testing images. Each image is a number between 0-9 and its size is 28x28 pixel RGB. In the same way that used in [6], the images are inverted before they enter as input to the first layer in our model, And therefore, the result of the digits appears in the white foreground on the backdrop of the black background. The reason is to use this method that the black background makes the edge detection more straightforward in digits recognition problems, now we come to the training process and we will use our suggested model to achieve higher accuracy with time-saving and during training stage using one cycle policy [24] to get faster training processes, then the model evaluated on test data that gathered from the dataset. The work environment in which the model was trained in the fastai framework [25] and python used as a programming language. After 20 cycles, our model achieved an accuracy 99.6, this result considered the best result obtained so far as shown in Table 1 which the proposed method was compared with the rest of the previous methods.

Table 1. Comparison of accuracy between the proposed method and previous methods

Method	Accuracy
“Handwritten Arabic numeral recognition using a multi-layer perceptron” [9].	93.8%
“Handwritten Arabic numeral recognition using deep learning neural networks” [12].	97.4%
“An Efficient Recognition Method for Handwritten Arabic Numerals Using CNN with Data Augmentation and Dropout” [16].	99.4%
Proposed method	99.6%

4. CONCLUSION AND FUTURE WORK

Arabic Handwritten Digits Recognition is An important and scalable area at this moment. Its applications are very important, especially in the Arab countries. This work is based on pre-trained Convolutional neural networks using the ResNet-34 Model and obtained a higher accuracy with 99.6%, and that is done by testing on MADBase Arabic handwritten digit dataset which contains 60000 training and 1000 testing images. The future work is focused on another pre-trained model such as VGG, Inception etc. to try to improve the result of recognition Arabic handwritten digits.

REFERENCES

- [1] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, "Handwritten digit recognition: benchmarking of state-of-the-art techniques," *Pattern Recognit.*, vol. 36, no. 10, pp. 2271–2285, 2003.
- [2] A. Rafalovitch and R. Dale, "United Nations general assembly resolutions: A six-language parallel corpus," in *Proceedings of the MT Summit*, vol. 12, pp. 292–299, 2009.
- [3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [4] M. M. Saufi, M. A. Zamanhuri, N. Mohammad, and Z. Ibrahim, "Deep learning for roman handwritten character recognition," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 12, no. 2, pp. 455–460, 2018. doi: 10.11591/ijeecs.v12.i2.pp455-460.
- [5] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv Prepr. arXiv1511.08458*, 2015.
- [6] T. Wiatowski and H. Bölcskei, "A mathematical theory of deep convolutional neural networks for feature extraction," *IEEE Trans. Inf. Theory*, vol. 64, no. 3, pp. 1845–1866, 2017.
- [7] J. Han and C. Moraga, "The influence of the sigmoid function parameters on the speed of backpropagation learning," in *International Workshop on Artificial Neural Networks*, pp. 195–201, 1995.
- [8] N. FatihahSahidan, A. Khairi Juha, N. Mohammad, and Z. Ibrahim, "Flower and leaf recognition for plant identification using convolutional neural network," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 16, pp. 737–743, 2019.
- [9] N. Das, A. F. Mollah, S. Saha, and S. S. Haque, "Handwritten Arabic Numeral Recognition using a Multi Layer Perceptron," *Proceeding National Conference on Recent Trends in Information Systems*, pp. 200–203, 2006.
- [10] A. Selwal and I. Raoof, "A Multi-layer perceptron based intelligent thyroid disease prediction system," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 17, no. 1, pp. 524–533, 2019. doi: 10.11591/ijeecs.v17.i1.pp524-532.
- [11] J. Li, J. Cheng, J. Shi, and F. Huang, "Brief introduction of back propagation (BP) neural network algorithm and its improvement," in *Advances in computer science and information engineering*, Springer, pp. 553–558, 2012.
- [12] A. Ashiquzzaman and A. K. Tushar, "Handwritten Arabic numeral recognition using deep learning neural networks," in *2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, pp. 1–4, 2017.
- [13] A. F. Agarap, "Deep learning using rectified linear units (relu)," *arXiv Prepr. arXiv1803.08375*, 2019.
- [14] D. M. Kline and V. L. Berardi, "Revisiting squared-error and cross-entropy functions for training neural network classifiers," *Neural Comput. Appl.*, vol. 14, no. 4, pp. 310–318, 2005.
- [15] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," in *ICML*, vol. 2, no. 3, pp. 7, 2016.
- [16] A. Ashiquzzaman, A. K. Tushar, A. Rahman, and F. Mohsin, "An efficient recognition method for handwritten arabic numerals using cnn with data augmentation and dropout," in *Data Management, Analytics and Innovation*, Springer, pp. 299–309, 2019.
- [17] S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: when to warp?," in *2016 international conference on digital image computing: techniques and applications (DICTA)*, pp. 1–6, 2016.
- [18] D. Pedamonti, "Comparison of non-linear activation functions for deep neural networks on MNIST classification task," *arXiv Prepr. arXiv1804.02763*, 2018.
- [19] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [20] L. Yang, S. Hanneke, and J. Carbonell, "A theory of transfer learning with applications to active learning," *Mach. Learn.*, vol. 90, no. 2, pp. 161–189, 2013.
- [21] A. K. Reyes, J. C. Caicedo, and J. E. Camargo, "Fine-tuning Deep Convolutional Networks for Plant Recognition," *CLEF (Working Notes)*, vol. 1391, pp. 467–475, 2015.
- [22] G. Rosa, J. Papa, A. Marana, W. Scheirer, and D. Cox, "Fine-tuning convolutional neural networks using harmony search," in *Iberoamerican Congress on Pattern Recognition*, pp. 683–690, 2015.
- [23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- [24] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1-learning rate, batch size, momentum, and weight decay," *arXiv Prepr. arXiv1803.09820*, 2018.
- [25] J. Howard and S. Gugger, "Fastai: A Layered API for Deep Learning," *Information*, vol. 11, no. 2, pp. 108, 2020.