

# Comparison of the efficiency of neural network algorithms in recognition and classification problems

Alexey Beskopylny<sup>1,\*</sup>, Alexandr Lyapin<sup>1</sup>, Nikita Beskopylny<sup>1</sup>, Elena Kadomtseva<sup>1</sup>

<sup>1</sup>Don state technical University, 1,Gagarin square, 344000, Rostov-on-Don, Russia

**Abstract.** The article is devoted to the problem of comparing the effectiveness of feedforward (FF) and convolutional neural networks (CNN) algorithms in the problems of handwritten digit recognition and classification. In recent years, the attention of many researchers to the FF and CNN algorithms has given rise to many hybrid models focused on solving specific problems. At the same time, the efficiency of each algorithm in terms of accuracy and labour intensity remains unclear. It is shown that in classical problems, FFs can have advantages over CNN in terms of labour intensity with the same accuracy of results. Using the handwritten digits data from the MNIST database as an example, it is shown that FF algorithms provide greater accuracy and require less computation time than CNN.

## 1 Introduction

Recently, artificial neural networks have received a lot of attention and have achieved outstanding results in various fields of research. Most of these algorithms involve identifying different relationships of objects of interest. Improvement of neural network algorithms has led to the emergence of a large number of software packages; however, the question of the effectiveness of these algorithms in solving similar problems has not yet been answered. From all the variety of algorithms, consider convolutional neural networks and feedforward networks.

Recently, the implementation of algorithms for convolutional neural networks (CNN) [1] has achieved significant results in various fields of research, such as visual recognition [2], search and recognition of defects [3,4], object classification [5], etc. By their very nature, CNNs are built on reasonably stringent assumptions that the input data must have a regular grid-like structure. On the one hand, such a restriction accelerates the convergence of the algorithm, on the other hand, it prevents the promotion and use of CNN for solving many problems in which data of an irregular structure is widespread. Direct linkage based object detection in classification problems generally does not perform well in complex scenarios, while deep neural networks potentially provide a powerful alternative. In [6], it is noted that the architecture of deep CNN (convolutional neural network) is immune to noise in image classification.

---

\* Corresponding author: [besk-an@yandex.ru](mailto:besk-an@yandex.ru)

Convolutional neural networks (CNNs) are a type of multilayer feedforward neural network that has achieved the most advanced results in visual recognition problems. Thanks to its multistage structure, CNNs can extract robust high-level image features, which is a complex problem in machine vision tasks. In [7], an unsupervised convolutional neural network (CNN) is proposed that uses principal component analysis to learn convolutional filters.

In [8], the authors proposed a CNN design based on the FF methodology. This algorithm offers an alternative approach to choosing CNN filter weights. A detailed comparison of the two design methodologies is carried out. The comparison shows that the accuracy of FF algorithms is lower than that of CNN, but many of the positive qualities of FF algorithms are not disclosed. The prospects for the development of algorithms are shown, in particular, as future extensions, it is advisable to develop a method aimed at increasing productivity in the problems of classification and countering hostile attacks. One possible direction is to use simpler networks with fewer parameters and then use the ensemble method to improve classification efficiency.

Feedforward neural network (FNN) is of great interest to many researchers in many applications due to its versatile approximating ability. The article [9] proposes a new FNN learning algorithm using the concept of sparse representation. The main advantage of the proposed algorithm is that it is able to train the original network and optimize the network structure simultaneously. The proposed algorithm consists of two main stages: structure optimization and weight update. During the structure optimization phase, the sparse representation method is used to select the critical hidden neurons that minimize the residual inference error. At the weight update stage, a dictionary-based learning method is implemented to update the network weights by maximizing the output diversity of hidden neurons. This weight update process is designed to improve design optimization efficiency. Based on several test classification and regression problems, the authors presented experimental results comparing the proposed algorithm with modern methods. The simulation results show that the proposed algorithm offers comparative performance in terms of final network size and generalizability.

FF algorithms fit well into more complex problems and find application in hybrid neural networks. A study [10] explored the possibility of using a feedforward neural network (FNN) to predict the axial bearing capacity of steel pipes filled with concrete. An evolutionary optimization (IWO) algorithm was used to tune and optimize the weights and biases of the FNN to build a hybrid FNN - IWO model and improve its predictive performance. The results showed that the FNN - IWO algorithm predicts axial load capacity with an accuracy of 0.979.

The above review shows that in many technical applications, the use of simple algorithms is often more justified due to their versatility, low sensitivity to noise and distortion. Thus, the purpose of this article is to compare the FNN and CNN algorithms in digit recognition and classification problems.

## 2 Materials and methods

One MNIST database was used to compare the FF and CNN algorithms in the problems of handwritten digit recognition. In deep learning models based on FF algorithms, a neuron is a weighted sum of all elements of the input vector  $\mathbf{w}$  using the weight vector  $\mathbf{x}$  (as well as the additive component of the displacement  $w_0$ ), and then the activation function  $\sigma$

$$y(\mathbf{x}, \mathbf{w}) = \sigma \left( w_0 + \sum_{i=1}^n \mathbf{w}_i \mathbf{x}_i \right) \quad (1)$$

where the ReLu function is used as the activation function

$$\sigma(z) = \max(0, z) \tag{2}$$

The well-known function of minimization of the mean square error was chosen as the loss function

$$E(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n \|\mathbf{y}_i - \hat{\mathbf{y}}_i\|^2 \tag{3}$$

where  $n$  is the number of training samples,  $\mathbf{y}_i$  and  $\hat{\mathbf{y}}_i$  multidimensional vectors, respectively, representing the desired and actual results corresponding to the input  $i$ -th sample.

The neural network learning algorithm is based on the assumption that the structure of the network (that is, the number of hidden neurons) is fixed in advance. A model with two hidden layers was used.

The quality of the model (prediction accuracy) was assessed based on the coefficient of determination

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \tag{4}$$

In convolutional neural network models, a feature map is obtained by reapplying the function to subregions of the entire image by convolving the input image with a linear filter, adding a bias term, and then applying a nonlinear function. The operation of convolution of two functions  $f(u)$  and  $g(u)$  in a discrete form

$$y(n) = \sum_{u=-\infty}^{\infty} f(u)g(n-u) \tag{5}$$

Optimization of weight functions is carried out using Adam's stochastic optimization method, implemented in the Keras module. CNN, designed for the handwriting recognition task, accepts  $n$ -digit input text. The input data vector is represented as  $\mathbf{W} \in R^{u \times v}$ , where  $u$  and  $v$  are the dimensions of the input data. Consequently, the CNN model will perform convolution operations on  $\mathbf{W}$  using the convolution filter  $j$ , whose weight has dimension  $\mathbf{X}^j \in R^{m \times k}$ . Using a filter  $\mathbf{X}^j$ , we can get a feature  $c_i$  generated from the digits window.

### 3 Results and discussions

The calculation results are shown in Table 1. Table 1 shows that for this example of handwritten digit recognition from the well-known MNIST database, the FF algorithm is less laborious in terms of computation time and gives an acceptable accuracy. The CNN algorithm gives good forecasting accuracy almost immediately, but a further increase in epochs leads

to a loss of accuracy and an increase in losses. This can be explained by the retraining of the system and taken into account in further calculations.

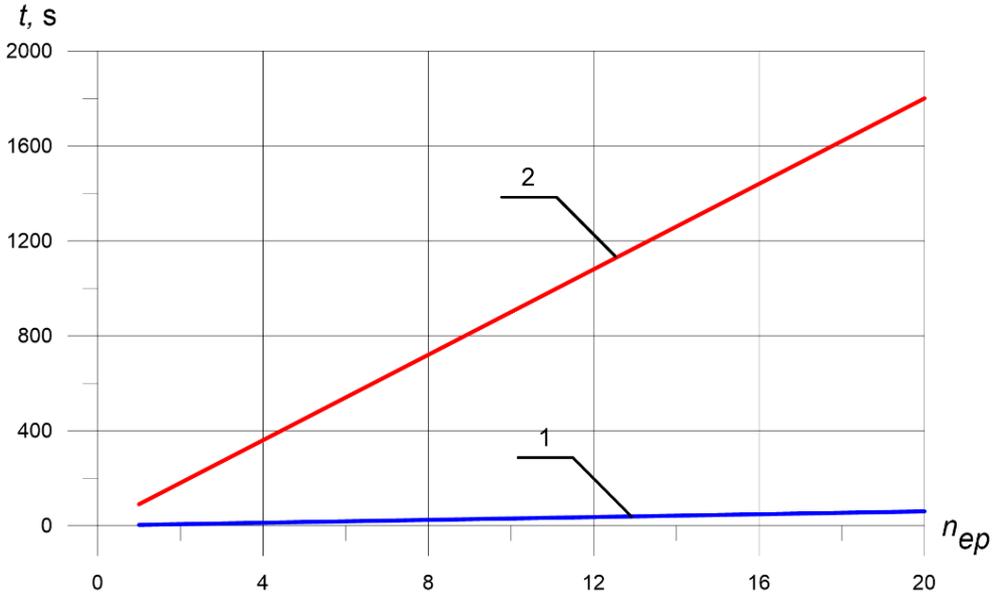
The problem of retraining a neural network is noted in various works. For example, in [11], three common issues are pointed out. First, complex algorithms prevent the dissemination of most of the data. The second is the retraining of the training data. Thirdly, it is the diversity in the distribution of data, since there are different prototype data in imbalanced datasets. To overcome this problem, all available data distributions to create realistic samples with less component in the training sample are used in the training process.

In articles [12, 13], to prevent overfitting, the dropout method is used, which in recent years has become the main regularizer for multilayer neural networks. The exclusion of individual data elements is carried out on the basis of a hierarchical mixing of experts, which corresponds to the tree hierarchy.

**Table 1.** Results of calculating the efficiency of algorithms

N	Epoch number	Algorithm CNN			Algorithm FF		
		Calculation time	Losses	Accuracy	Calculation time	Losses	Accuracy
1	1	90	0.0797	0.9731	3	0.0920	0.9730
2	2	180	0.0869	0.9745	6	0.0769	0.9763
3	3	270	0.0990	0.9766	9	0.0675	0.9797
4	5	450	0.1405	0.9726	15	0.0828	0.9785
5	7	630	0.1519	0.9771	21	0.0780	0.9797
6	10	900	0.1629	0.9791	30	0.0839	0.9810
7	15	1350	0.2731	0.9799	45	0.0958	0.9827
8	20	1800	0.3906	0.9775	60	0.1036	0.9818

Figure 1 shows the dependences of the complexity of the FF and CNN algorithms in the form of computation time.



**Fig. 1.** Dependence of the counting time on the number of epochs: 1 - FF algorithm; 2 - CNN algorithm.

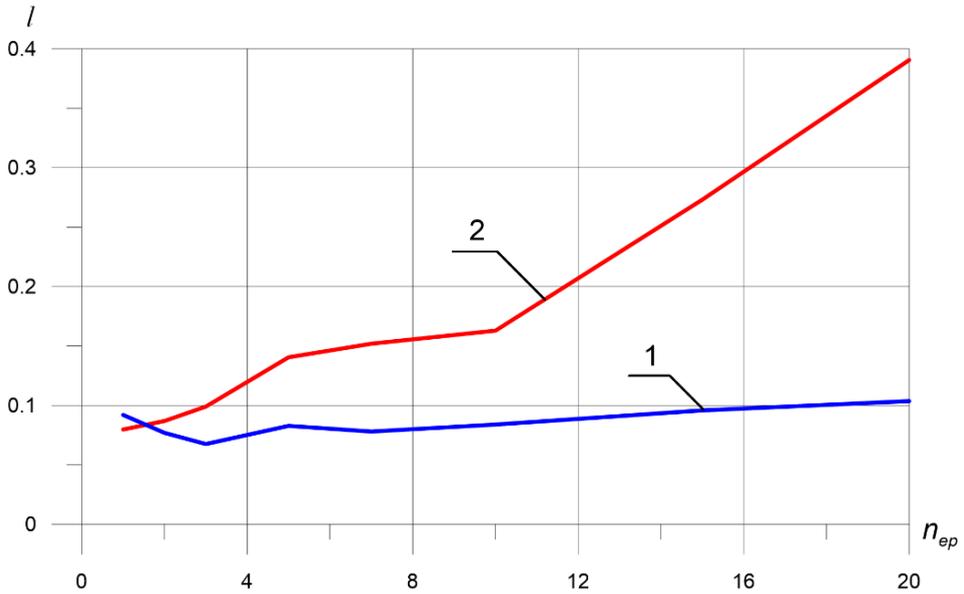
It can be seen (Figure 1) that the FF algorithm requires significantly less computation time compared to convolutional network technologies.

The low complexity and versatility of the FF algorithm allow it to be effectively used in technical applications when the initial data are irregular structures. For example, in [14], when monitoring and diagnosing underground systems by dynamic methods, [15] during detection and localization in plate-like structures, the initial signal is a complex dependence, which contains a high noise level due to the impact of the dynamics of passing vehicles or impact.

The temporal information contained in the waveform is vital for localizing the damage. Therefore a time-varying function is introduced that stores the temporal information to improve the accuracy of localizing the damage. Using a one-dimensional convolutional neural network to correlate a time-varying function with an injury site directly can be very useful.

Another example of the application of neural network algorithms is the prediction of deep failure due to fatigue damage in highly loaded gears. It was shown in [16] that the appearance of fatigue cracks occurs under the surface and can lead to a breakdown of the mechanism. The construction of effective neural networks will allow detecting deep damage at an early stage.

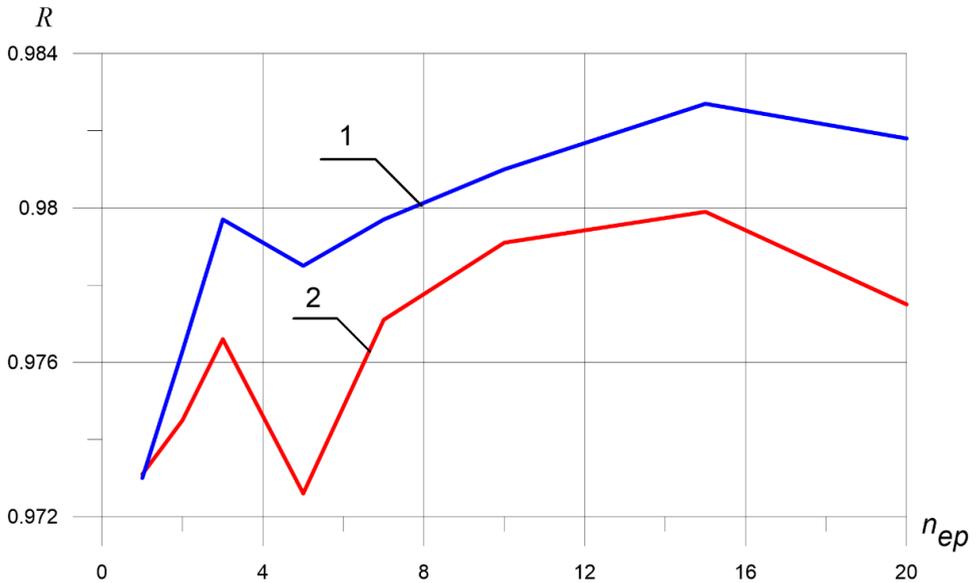
Figure 2 shows the dependence of losses on the number of epochs.



**Fig. 2.** Dependence of losses on the number of epochs: 1 - FF algorithm; 2 - CNN algorithm.

It is seen that the losses in the implementation of the FF algorithm are relatively stable, while in the implementation of the CNN algorithm, the losses increase sharply after 10 epochs.

Figure 3 shows the dependences of the prediction accuracy of the FF and CNN models.



**Fig. 3.** Dependence of the prediction accuracy on the number of epochs: 1 - FF algorithm; 2 - CNN algorithm.

Figure 3 shows that the accuracy of the FF algorithm is higher than that of CNN. It is also seen that both algorithms experience the problem of overfitting after 15 epochs and further training is not effective. At the initial stage of training, when the number of epochs is 1, the CNN method gives a slightly more accurate result. But, further training using the FF method

leads to an increase in accuracy, while the CNN algorithm shows consistently lower accuracy.

## 4 Conclusions

The variety of deep learning methods leads to the need to evaluate the effectiveness of algorithms and identify negative factors affecting the recognition process. The article compares feedforward algorithms FF and CNN convolutional neural networks and evaluates their efficiency in terms of labour intensity (computation time) and forecast accuracy. It is shown that, on the basis of the known MNIST base, the FF algorithm is more efficient in comparison with CNN both in computation time and in forecast accuracy.

The results of calculations for both algorithms showed that both networks experience an overfitting problem after 15 training epochs. Overcoming the difficulties associated with the retraining of the network is beyond the scope of this study, but when compared with the data of other researchers [11-13,17,18], it is shown that the solution of such a problem is possible on the basis of regularization methods based on screening out individual data.

## References

1. Bai S, Zhang F and Torr P 2021 *Pattern Recognition* **110** 107637
2. He L, Xi D, Li J and Zhu J A 2020 *Applied Sciences* **10** 5809
3. Lin Z, Ye H, Zhan B and Huang X 2020 *Applied Sciences* **10** 6085
4. Fan Z, Li C, Chen Y, Wei J, Loprencipe G, Chen X and Di Mascio P 2020 *Materials* **13** 2960
5. Beskopylny A, Lyapin A, Anysz H, Meskhi B, Veremeenko A, Mozgovoy A 2020 *Materials* **13** 2445
6. Meng Z, Zhang M, Wang H 2020 *Sensors* **20** 4974
7. Qaraei M, Abbaasi S, Ghiyasi-Shirazi K 2021 *Information Sciences* **545** 241–253
8. Kuo C J, Zhang M, Li S, Duan J, Chen Y J 2019 *Journal of Visual Communication and Image Representation* **60** 346–359
9. Yang J, Ma J 2019 *Expert Systems with Applications* **116** 255-264
10. Nguyen H, Ly H, Tran V, Nguyen T, Le T, Pham B 2020 *Materials* **13** 1205
11. Zareapoor M, Shamsolmoali P, Yang J 2021 *Mechanical Systems and Signal Processing* **149** 107175
12. Irsoy O, Alpaydin E 2021 *Neurocomputing* **419** 148-156
13. Elleuch M, Kherallah M 2020 *Advances in Intelligent Systems and Computing* 2020 **1179** 103–112
14. Lyapin A, Beskopylny A, Meskhi B 2020 *Sensors* **20** 5241
15. Zhang S, Li C, Ye W 2021 *Mechanical Systems and Signal Processing* **147** 107107
16. Beskopylny A, Meskhi B, Onishkov N, Kotelnitskaya L, Ananova O 2020 *Metals* **10** 600
17. Solov'ev A N, Cherpakov A V, Vasil'ev P V, Parinov I A, Kirillova E V 2020 *Advanced Engineering Research* **20(3)** 205-215 <https://doi.org/10.23947/2687-1653-2020-20-3-205-215>
18. Zhilin V V Safar'yan O A 2020 *Vestnik of Don State Technical University* **20(2)** 196-

200 <https://doi.org/10.23947/1992-5980-2020-20-2-196-200>