Science & Technology Development Journal – Engineering and Technology 2024, ():1-8

Open Access Full Text Article

Research Article

Convolutional neural networks for image object recognition and classification with large-scale and complex data

P. T. Anh[1], N. K. Diep[1,*], N. V. Trong[2]

ABSTRACT

The purpose of this study is to investigate the application of artificial neural networks for image recognition and classification tasks. Additionally, the research aims to explore the usage of convolutional neural networks in handling large-scale and complex data. These networks are designed to reduce memory storage requirements and hierarchically extract and aggregate features from input data, which is essential for efficient data processing. The research employs the use of convolutional neural networks and large-scale and complex data processing techniques, which are at the forefront of advancements in machine learning. The neural network is constructed and trained with large-scale and complex data, ensuring that the system is robust and capable of handling real-world datasets. The performance is evaluated using various configurations of convolutional and pooling layers, which are integral components of CNNs that help in feature detection and reduction of computational complexity. The study also involves the development of a library of methods based on the ``.NET Standard 2.0'' platform, which is a widely-recognized framework for building high-performance applications. Additionally, a window application using ``.NET 6.0'' and ``WPF'' platforms is developed, demonstrating the practical implementation of the research. The study explores the quality of the proposed convolutional neural network based on different configurations of convolutional and pooling layers and the size of the convolution filter. The best results are achieved when using 3 blocks of convolutional and pooling layers with a filter size of 3 x 3 pixels. The network achieves optimal accuracy in image object classification after being trained for 14 epochs. The findings demonstrate the effectiveness of the proposed convolutional neural network architecture and its ability to handle large-scale and complex data efficiently for image recognition and classification tasks. The study's outcomes contribute significantly to the field of computer vision and offer a promising direction for future research in neural network optimization for image processing.

Key words: convolutional neural networks, image classification, data, artificial neural networks, deep learning

[1]*Institute of Information Technology, AMST, Ho Chi Minh City, Vietnam*

[2]*Institute of Control Sciences V.A. Trapeznikov Russian Academy of Sciences, Moscow, Russia*

Correspondence

N. K. Diep, Institute of Information Technology, AMST, Ho Chi Minh City, Vietnam

Email: diepst62@mail.ru

History

- Received: 31-8-2023
- Accepted: 29-3-2024
- Published Online:

DOI :

Check for updates

INTRODUCTION

Recognition and classification of image objects have become common tasks today, and they are being increasingly well-solved through the utilization of Artificial Neural Networks (ANNs). ANNs are one of the most extensively studied branches of artificial intelligence development and find broad applications in various fields such as medicine, economics, information processing, computer vision, and more. In image recognition and classification tasks, handling large amounts of data poses two challenges: firstly, selecting the optimal neural network model, and secondly, constructing an efficient algorithm for its training.

A significant contribution to the advancement of ANNs for image object classification was the creation of the ImageNet project. It constitutes an extensive visual database comprising over 14 million manually labeled images, designed for research on software used in image object recognition[1]. Additionally, it is noteworthy that modern approaches to image object recognition and classification commonly employ Convolutional Neural Networks (CNNs)[2]. High performance and accuracy in classifying image objects from the ImageNet dataset can be achieved through the use of the deep traditional CNNs architecture[3], which was proposed back in 1989 for recognizing handwritten digits[4].

The aim of this article is to investigate the use of CNNs in the tasks of recognition and classification of image objects within large-scale and complex data. To conduct this research, a software application was developed for constructing CNNs training them, and testing their performance accuracy.

The paper is organized as follows. Section 2 introduces the theoretical foundations of ANNs, CNNs,

Cite this article : Anh P T, Diep N K, Trong N V. **Convolutional neural networks for image object recognition and classification with large-scale and complex data**. *Sci. Tech. Dev. J. – Engineering and Technology* 2024; ():1-8.

and large-scale and complex data processing techniques. Section 3 describes the methods, simulation tools, and results of the proposed CNN for image object classification. Section 4 concludes the paper and discusses future directions.

PROBLEM

Theoretical Foundations

ANNs are computational mathematical models that allow obtaining desired output data from a set of input data. They simulate the biological processes occurring in the human nervous system.

The key elements of the nervous system are a collection of nerve cells called neurons. The body of a neuron is called the soma. The neuron has dendrites that receive information and an axon through which it transmits information to other neurons in the nervous system via synaptic connections[5]. Each neuron can have multiple dendrites and only one axon, which carries impulses to the next neurons in the nervous system through synapses.

It is precisely the biological neurons that inspired the creation of artificial ones. Let's construct a simplified model of an artificial neuron (Figure 1) and describe its operation in more detail.

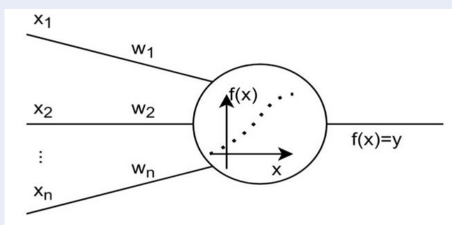


Figure 1: Simplified model of an artificial neuron.

The depicted artificial neuron (Figure 1) has dendrites with synapses n having weights w$_i$, $i = \overline{1,...,n}$, represents the incoming signals received at the synapses, and f(x) is a certain transfer function or threshold activation function, y represents the output signal. When the synaptic inputs are received, the incoming signals are scaled according to their weight coefficients and fed into the summator, which processes them according to the formula:

$$x = \sum_{i=1}^{n} x_i w_i \tag{1}$$

where x is the argument of the threshold activation function; n is the number of dendrites; x_i is the input signals; w_i is the synaptic weight coefficients. The threshold activation function is represented as:

$$y = f(x) \tag{2}$$

where y is the output signal of the neuron's axon.

The choice of activation function depends on the desired output signal and the performance of the ANN, as well as the nature of the problem being solved. The sigmoid function is the most commonly used activation function, which can be discrete or continuous[6]. An example of the simplest activation function is the Heaviside step function (Figure 2, a), which is a discrete sigmoid with a parameter and is defined as follows:

$$y = \begin{cases} 1 \; where \; x \geq a, \\ 0 \; where \; x < a \end{cases} \tag{3}$$

where a is a certain parameter. Quite often, the sigmoidal logistic function with parameters (Figure 2, b) is used, which is continuous, nonlinear, and well-suited for image object classification tasks. It is described by the function:

$$y = \frac{b}{c + e^{dv}} \tag{4}$$

where b, c, d are parameters of the function. Another widely used activation function is the "ReLU" (rectified linear unit) function (Figure 2, c), which returns the input value when it is positive and 0 when it is negative:

$$y = \begin{cases} 0 \; where \; x < 0, \\ x \; where \; x \geq 0 \end{cases} \tag{5}$$

There are numerous variations in constructing ANNs, distinguished by their structures - models of neuron connections. Hierarchical, recurrent, and competitive structures are among the types. Overall, all models of ANNs are constructed similarly. Neurons are arranged in layers within the network. Initially, information is fed into the neurons of the input layer, which then pass it on to the neurons of the so-called hidden layer. The primary data processing occurs in the hidden layers, after which the information is forwarded to the output layer. The number of neurons and hidden layers is chosen depending on the specific task that the neural network is designed to solve, the volume of data, and the available computational resources.

The perceptron developed by Rosenblatt is considered the first ANN, based on the model of the brain's operation in recognizing optical patterns[7]. Structurally, it is a simple feedforward network. In modern terminology, such an ANN belongs to single-layer networks with a hierarchical structure.

As the volume of data increases, the performance and scalability of ANNs become a significant concern. Big Data often presents challenges in terms of

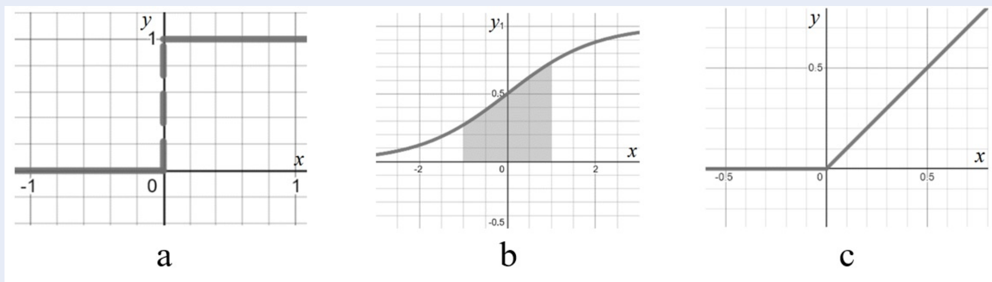


Figure 2: Threshold activation functions: a - discrete sigmoid with parameter a = 0, b – sigmoidal logistic function, c – ReLU function

handling massive datasets, processing complex patterns, and training large-scale neural networks efficiently. The sheer size and complexity of Big Data can lead to increased computational demands and potentially hinder real-time or near-real-time processing. To address these challenges, specialized hardware, distributed computing, and advanced optimization techniques are often employed to effectively utilize the available computational resources and achieve acceptable processing times.

The model of a neural network, its layers, activation functions, and learning algorithms should be chosen based on the specific task at hand. For building the proposed neural network, a classical task was selected - image object classification. Partly, this is due to the availability of Big Data with images for testing CNNs[8]. The neural network will determine whether a loaded image contains the face of a man or a woman. Due to the format of the input data (an array of pixel values), using traditional fully connected neural networks is impractical due to the need for an enormous number of neurons in the input layer. CNNs are the better choice here, especially because of the convolution operation, which reduces the amount of information stored in memory and hierarchically extracts and aggregates features from the input data. This increases the computational power of the neural network and its ability to model more complex dependencies.

Experiment reparation

The proposed CNN architecture consists of several blocks of convolutional and pooling layers, a flattening layer, fully connected neuron layers, and an output neuron. Convolutional and pooling layers are essential for detecting general image features. The fully connected neuron layers are used to identify features specific to male or female faces. There is a single initial layer that constitutes a fully connected neuron. The threshold activation function chosen for all neurons, except the initial one, is the aforementioned ”ReLU” function, which allows saving computational resources by not calculating values less than 0. The output neuron uses the sigmoid activation function. The schematic model of this neural network is shown in Figure 3.

When dealing with large-scale and complex data in the context of CNNs, one of the main challenges is the sheer volume of data. Training and processing large-scale CNNs with massive datasets can be computationally expensive and time-consuming. This requires specialized hardware and distributed computing techniques to efficiently handle the data and ensure reasonable processing times. Additionally, the need for extensive computational resources can lead to scalability issues, where processing times increase significantly with the growth of the dataset. Furthermore, as the volume of data increases, the risk of overfitting the model may also arise, necessitating careful regularization techniques and model evaluation to avoid potential performance degradation on new, unseen data.

METHODS

Input Data

Input data for the CNN typically consists of images or image-like data. Each image serves as an input to the network and is represented as a matrix of pixel values. The dimensions of the image matrix depend on the resolution and color channels of the image.

For grayscale images, the matrix will have two dimensions (height and width), where each pixel value represents the intensity of the corresponding pixel in the image. The intensity value is typically represented as an integer between 0 and 255, where 0 indicates black and 255 indicates white.

For color images, the matrix will have three dimensions (height, width, and channels), where each pixel value represents the intensity of each color channel

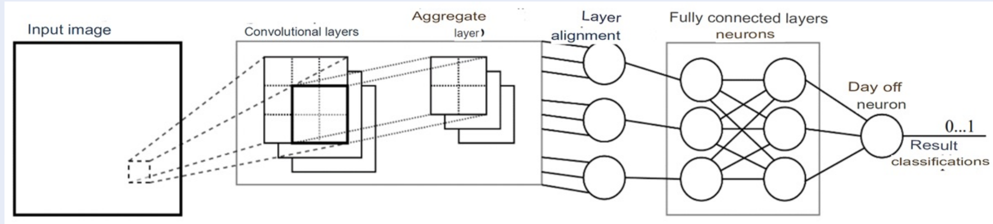


Figure 3: The schematic model of the developed CNN for image object classification.

(red, green, and blue) in the corresponding pixel. The intensity values for each color channel are also typically represented as integers between 0 and 255.

Before feeding the images into the CNN, preprocessing steps may be applied, such as normalization to scale the pixel values to a certain range (e.g., [0, 1]) or standardization to make the data have zero mean and unit variance. Additionally, data augmentation techniques might be used to increase the diversity of the training data by applying random transformations (e.g., rotations, flips, and translations) to the images.

The input data is essential for training the CNN to learn relevant features and patterns in the images, which enable it to perform tasks like image recognition, classification, or object detection.

Methods, imulation Tools

For building the proposed neural network, its training, and testing, ”TensorFlow” was chosen. It is an open-source library designed for constructing and training neural networks, developed, maintained, and supported by Google. For the convenience of developing the CNN, the high-level API ”Keras.NET” [9] was additionally used. To manage Keras.NET, a library of methods was developed in the C# programming language, using the .NET Standard 2.0 development platform.

For visually displaying the processes' settings performed within the mentioned library, a window application was developed using the C# programming language and the .NET 6.0 and WPF development platforms. To construct the graph showing the classification accuracy dependency on the training epochs, the OxyPlot library [10] was employed.

The application can be roughly divided into three parts: building the CNN, its training, and testing.

RESULT AND DISSCUSION

During the construction of the neural network, the user is provided with the option to choose the width and height to which the image will be compressed, as well as the initial parameters of the CNN, such as learning rate and momentum. CNN elements are added in blocks, consisting of convolutional and pooling layers. The user can select the number of convolutional and fully connected layers, as well as the sizes of the convolutional and pooling layers. The activation function can be either ”ReLU” or sigmoid. The interface displaying these application features is depicted in Figure 4.

The next part is the training of the neural network. For this, it is necessary to first select a dataset of images, divided into 2 folders corresponding to the classification objects, in this case, ”male” and ”female.” To perform automatic testing of the CNN, a similarly distributed dataset should be chosen. Next, the number of epochs should be specified, and the ”Start Training” button is pressed. During the training process, a graph showing the CNN's accuracy dependency on the training epochs is displayed. The obtained neural network can be saved. The described interface is shown in Figure 5, a.

For manual testing of the neural network, the first step is to select an image. A rectangle will be displayed on the image, and it needs to be placed over the person's face. It is possible to test the performance of either the newly built CNN or a saved one loaded from the respective file. The interface for manual testing is shown in Figure 5, b.

Using the developed application, a study of the CNN's quality was conducted based on the number of blocks and sizes of the convolutional filter (Table 1). The number of blocks varied from 1 to 4, and the sizes of the convolutional filter ranged from 3 x 3 to 4 x 4 pixels. The number of training epochs was chosen to be slightly excessive - 20 epochs.

The data was divided into 2 subsets: one was used for training, and the other for testing after each epoch. It should be noted that the testing data was not used during the training process. Analyzing the obtained results (Table 1), it can be concluded that increasing the size of the convolutional filter to 4 x 4 pixels leads to a

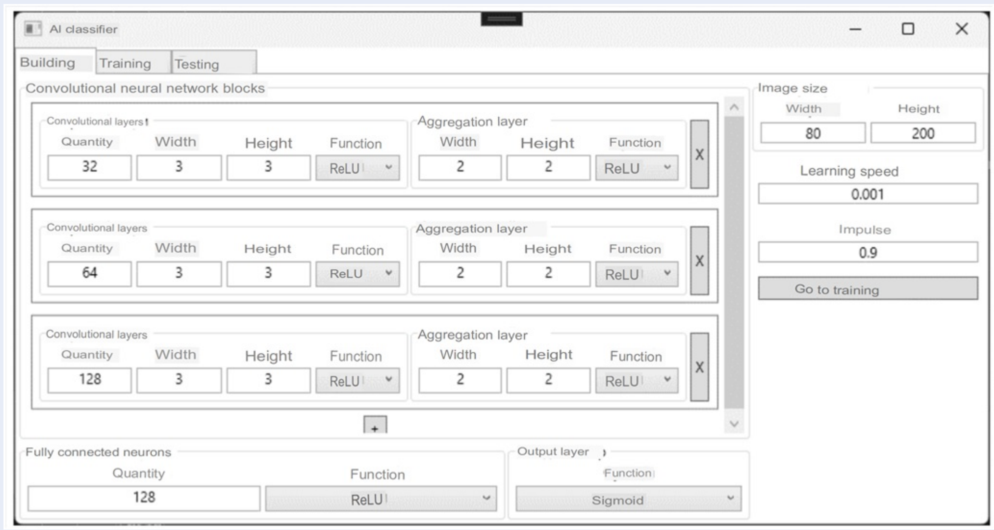
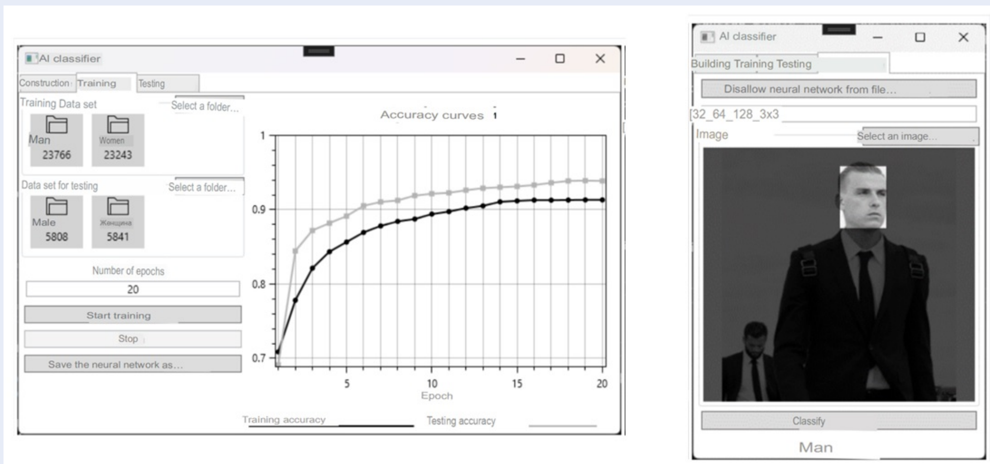


Figure 4: Configuration of building an artificial neural network



a b

Figure 5: Screenshots of the application working with the CNN using a convolutional filter of size 3 x 3 pixels and 3 blocks of convolutional and pooling layers and 3 blocks of convolutional and aggregating layers: a – training of the SNM, b – manual testing.

decrease in classification accuracy. This is due to the fact that increasing the size of the convolutional filter negatively affects the correct extraction of certain image features. Additionally, increasing the size of the convolutional filter reduces the depth of the CNN, which in turn reduces the classification accuracy. For the considered task, choosing the number of CNN blocks greater than 3 is not practical, as the training cost of the network increases significantly, while the classification accuracy is not substantially improved.

Based on these conclusions, let's analyze the accuracy graphs of the neural network training and testing with exactly 3 blocks of convolutional and pooling layers and a filter size of 3 x 3 pixels (Figure 5a). The optimal accuracy of image classification for the CNN is achieved when it is trained for 14 epochs. Further increasing the number of training epochs is not advisable.

In the article have been improved the methods for training neural networks to solve image compression tasks have been improved. A competitive neural net-

Table 1: Comparison of CNN Performance with Different Numbers of Blocks and Various Sizes of the Convolutional Filter

Number of Blocks	Filter, pixels	Training Accuracy, %	Testing Accuracy, %
1	3 x 3	77,8	81,1
1	4 x 4	74,9	80,6
2	3 x 3	86,0	87,2
2	4 x 4	86,6	86,2
3	3 x 3	91,3	93,3
3	4 x 4	89,6	90,4
4	3 x 3	91,4	92,7
4	4 x 4	90,9	92,0

work that uses image fragments as input signals is proposed. Additionally, information being processed is further analyzed in real-time using the Kalman-Mein filter. The test data showed that the neural network model is working optimally, and the next step is to launch the system on large-scale and complex. Several approaches were used for the image segmentation operator:

Forming a new generation only from offspring.

Employing the principle of elitism, where a certain portion of the parents is guaranteed to transition to the new population.

Replacing identical chromosome structures in the new generation. The genetic operators and approaches described above were utilized in the instrumental sound classification system to optimize its architecture and hyperparameters.

Computational experiments are conducted using the Urban8k dataset from the surveillance camera archive. The formulas below contain the obtained dissimilarity matrices when studying the influence of the number of layers in the convolutional neural network, which is part of the hybrid system, on classification accuracy. The dissimilarity matrix of a single Conv2D convolutional layer is as follows (Figure 6):

Analyzing the matrices, it can be concluded that the obtained values indicate satisfactory prediction accuracy. Specifically, for the last class, the number of correctly predicted classes is 15, while the number of incorrect predictions is 2. It should be noted that incorrect predictions were made for the second class. The dependence of error and training accuracy functions on the training epochs for the respective neural network topologies is shown in Figure 7.

The results of comparing these functions for one and two convolutional layers indicate a smoother practical convergence of the classifier, specifically for a single convolutional layer, with fixed other hyperparameters. Using the classifier corresponding to Figure 7 could lead to unstable neural network performance in practice. Meanwhile, the accuracy of the first classifier is 95%, and the second one is nearly 94%. Therefore, we can conclude that a single accuracy value is sometimes insufficient for a comprehensive evaluation of the neural model's effectiveness.

CONCLUSION

The proposed convolutional neural network can be successfully applied in another ANN designed, for example, to detect and localize multiple classification objects in an image, to perform face recognition or facial expression recognition. Additionally, such a CNN can be utilized as a discriminator within a generative adversarial neural network (GAN) setup, which will be explored further.

When dealing with large-scale and complex in the context of the proposed CNN, the volume and complexity of the data can present significant challenges. Training a CNN on large-scale datasets may require substantial computational resources and time. The need for extensive data preprocessing and augmentation techniques might arise to ensure sufficient training data for optimal performance. Furthermore, as the size of the dataset increases, it can become more challenging to maintain a balance between training accuracy and model generalization, leading to potential overfitting issues. Addressing these challenges might involve employing distributed computing and specialized hardware to cope with the high computational demands and enhance the efficiency of training and testing processes. Additionally, careful model evaluation and regularization techniques are essential to ensure that the CNN performs well on new, unseen data and avoids potential overfitting.

$$\begin{pmatrix} 47 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 48 & 0 & 1 & 3 & 0 & 0 & 1 & 0 \\ 0 & 0 & 41 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 35 & 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 2 & 47 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 42 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 48 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 43 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 15 \end{pmatrix}$$

Matrix 1

$$\begin{pmatrix} 47 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 46 & 0 & 1 & 1 & 0 & 0 & 2 & 1 \\ 1 & 0 & 41 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 1 & 0 & 36 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 1 & 3 & 43 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 42 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 48 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 42 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 15 \end{pmatrix}$$

Matrix 2

Figure 6: The matrixes of the developed CNN for image object classification.

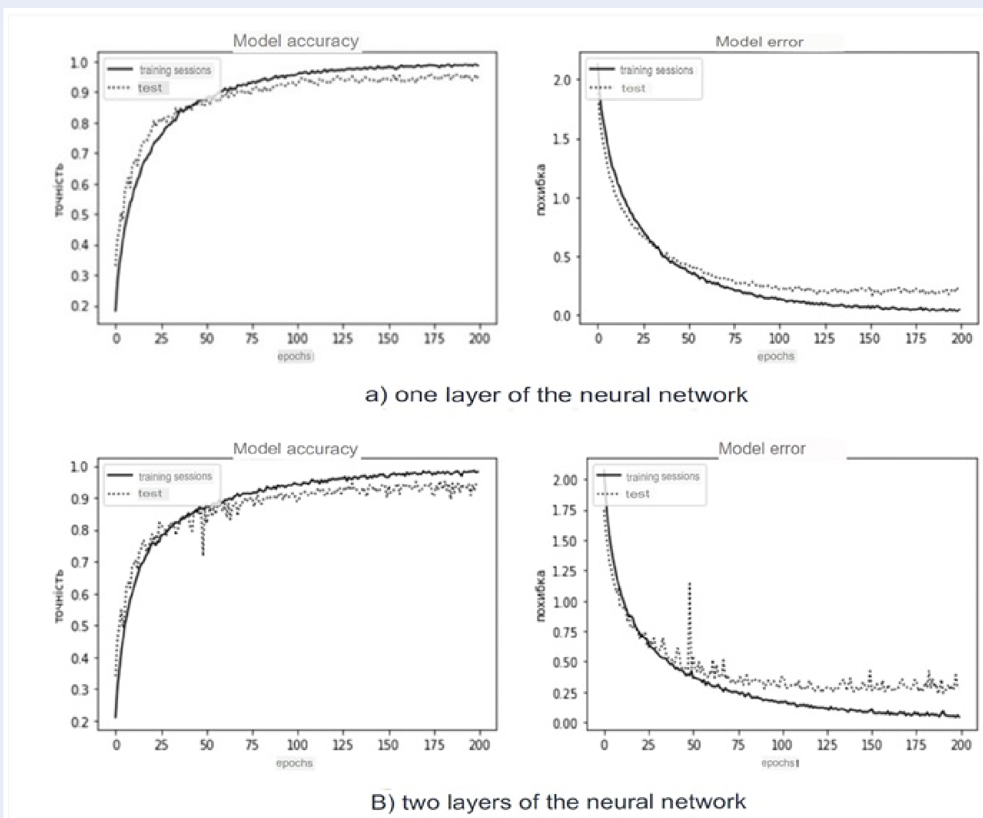


Figure 7: Accuracy and prediction error of models with different numbers of Conv2D layers

LIST OF ABBREVIATIONS

ANNs: Artificial Neural Networks

API: Application Programming Interface

CNNs: Convolutional Neural Networks

CONFLICT OF INTEREST

The authors confirm that there is not any conflict of interest related to the content reported in this paper.

AUTHORS' CONTRIBUTION

P. T. Anh is the main writer of the paper, who has excellent English writing and expression skills. He has written the introduction, literature review, methodology, results and discussion sections of the paper, using clear and concise language and appropriate academic style. He wrote the paper in clear and concise English, following the journal's guidelines and academic style.

N. K. Diep is the main researcher of the paper, who has deep research and idea generation and editing skills. He has proposed the research problem, objectives, questions and hypotheses, and edited and revised the paper.

N. V. Trong is the main programmer of the paper, who has good programming, testing and experimentation skills. He has implemented the convolutional neural network and the window application, performed the experiments and analyzed the results.

REFERENCES

1. Reynolds M. New computer vision challenge wants to teach robots to see in 3D. New Scientist [Internet];Available from: https://cutt.ly/sKlw30e.
2. Krizhevsky A, Sutskever I, Hinton EG. Advances in Neural Information Processing Systems. Communications of the ACM. 2017;60(6):84-90;Available from: https://doi.org/10.1145/3065386.
3. Adhikari T. Designing a Convolutional Neural Network for Image Recognition: A Comparative Study of Different Architectures and Training Techniques;.
4. Karrach L, Pivarciova E. Using a Convolutional Neural Network for Machine Written Character Recognition;PMID: 37754952. Available from: https://doi.org/10.21203/rs.3.rs-2712626/v1.
5. Chen Y. Research on Convolutional Neural Network Image Recognition Algorithm Based on Computer Big Data. Journal of Physics: Conference Series;Available from: https://doi.org/10.1088/1742-6596/1744/2/022096.
6. Sychugov et al. Application of Neural Networks for Object Recognition in Railway Transportation. In: Proceedings of Petersburg Transport University, Moscow, Russia; 2023;Available from: https://doi.org/10.20295/1815-588X-2023-2-478-491.
7. Desai C. Object Detection and Recognition of A Handwritten Hindi Character On a Background Image. 2023;.
8. Chaunah A. Gender Classification Dataset. 2020. [Online];Available from: https://cutt.ly/NKleYG3.
9. Karra J. Deep learning using Keras.Net with C#. [Online];Available from: https://cutt.ly/4KleSlY.
10. Reznykov I. Wpf application with real-time data in OxyPlot charts. [Online];Available from: https://svitla.com/blog/wpf-application-with-real-time-data-in-oxyplot-charts.

Tạp chí Phát triển Khoa học và Công nghệ – Engineering and Technology 2024, ():1-1

Open Access Full Text Article

Bài nghiên cứu

Mạng nơ-ron tích chập trong các bài toán nhận dạng và phân loại đối tượng hình ảnh với dữ liệu quy mô lớn và phức tạp

Phạm Tuấn Anh[1], Nguyễn Khắc Điệp[1,*], Nguyễn Văn Trọng[2]

TÓM TẮT

Mục đích nghiên cứu của bài báo là khảo sát việc áp dụng mạng nơ-ron nhân tạo cho các bài toán nhận dạng và phân loại ảnh. Ngoài ra, bài báo cũng nghiên cứu về việc áp dụng mạng nơ-ron tích chập để xử lý dữ liệu quy mô lớn và phức tạp. Những mạng này được thiết kế để giảm yêu cầu về bộ nhớ lưu trữ và trích xuất cũng như tổng hợp các đặc trưng từ dữ liệu đầu vào một cách có hệ thống, điều này là cần thiết cho việc xử lý dữ liệu một cách hiệu quả. Nghiên cứu sử dụng mạng nơ-ron tích chập và các kỹ thuật xử lý dữ liệu quy mô lớn và phức tạp, là những kỹ thuật tiên tiến trong lĩnh vực học máy. Mạng nơ-ron được xây dựng và huấn luyện với dữ liệu quy mô lớn và phức tạp, đảm bảo cho hệ thống mạnh mẽ và có khả năng xử lý các bộ dữ liệu thực tế. Hiệu suất được đánh giá sử dụng các cấu hình khác nhau của các lớp tích chập và pooling, những thành phần không thể thiếu của CNN giúp phát hiện đặc trưng và giảm độ phức tạp tính toán. Nghiên cứu cũng đã giới thiệu việc phát triển thư viện các phương pháp dựa trên nền tảng ".NET Standard 2.0", là một framework phổ biến trong việc xây dựng các ứng dụng hiệu suất cao. Bên cạnh đó, ứng dụng Windows sử dụng nền tảng ".NET 6.0" và "WPF" cũng được phát triển để triển khai nghiên cứu thực tế. Chất lượng của mạng nơ-ron tích chập được đề xuất dựa trên các cấu hình khác nhau của các lớp tích chập và pooling và kích thước của bộ lọc tích chập. Kết quả tốt nhất đạt được khi sử dụng 3 khối của các lớp tích chập và pooling với kích thước bộ lọc 3 x 3 pixel. Mạng đạt được độ chính xác tối ưu trong phân loại đối tượng hình ảnh sau khi được huấn luyện trong 14 epoch. Các kết quả cho thấy hiệu quả của kiến trúc mạng nơ-ron tích chập được đề xuất và khả năng xử lý dữ liệu quy mô lớn và phức tạp với các bài toán nhận dạng và phân loại ảnh. Kết quả của nghiên cứu đóng góp đáng kể cho lĩnh vực thị giác máy tính và mở ra nhiều triển vọng cho nghiên cứu tương lai trong việc tối ưu hóa mạng nơ-ron cho việc xử lý hình ảnh.

Từ khoá: mạng tích chập, phân loại ảnh, dữ liệu, mạng nơ-ron nhân tạo, học sâu

[1]*Viện Công nghệ thông tin, AMST, TP. Hồ Chí Minh, Việt Nam*

[2]*Viện Khoa học điều khiển V.A. Trapeznikov, Viện Hàn lâm khoa học Nga*

Liên hệ

Nguyễn Khắc Điệp, Viện Công nghệ thông tin, AMST, TP. Hồ Chí Minh, Việt Nam

Email: diepst62@mail.ru

Lịch sử

- Ngày nhận: 31-8-2023
- Ngày chấp nhận: 29-3-2024
- Ngày đăng:

DOI :

Trích dẫn bài báo này: Anh P T, Điệp N K, Trọng N V. **Mạng nơ-ron tích chập trong các bài toán nhận dạng và phân loại đối tượng hình ảnh với dữ liệu quy mô lớn và phức tạp**. *Sci. Tech. Dev. J. - Eng. Tech.* 2024; ():1-1.