ISSN 2278-3091
Volume 9, No.4, July – August 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse371942020.pdf
https://doi.org/10.30534/ijatcse/2020/371942020

Deep Neural Network Elements and their Implementation in Models of Protective Forest Stands with the Participation of Shrubs

Aliya Sh. Huzhahmetova[1], Alexandra V. Semenyutina[2], Victoria A. Semenyutina[3]
[1] Federal Scientific Center for Agroecology, Complex Melioration and Protective Afforestation of the Russian Academy of Sciences, Volgograd, Russia
[2] Federal Scientific Center for Agroecology, Complex Melioration and Protective Afforestation of the Russian Academy of Sciences, Volgograd, Russia
[3] Federal Scientific Center for Agroecology, Complex Melioration and Protective Afforestation of the Russian Academy of Sciences, Volgograd, Russia

ABSTRACT

The article provides a scientific analysis of the possibilities of using shrubs and their models for adaptive and landscape arrangement of territories.Objects of research: species and forms of the genus Corylus L. (Corylus avellana L., forms of S. pontica C. Koch. ‘ 'President',' Futkurami’, 'Circassian-2'). Age 20 years, grow in conditions of chestnut soils. Polygon-cadastral number 34:34:000000:122. We used a morphological method for determining biometric values and determining deviations from the optimal size, and an electronic-physiological method (s230kit conductometer, Dualex Scientific device) for diagnostics of state parameters.Currently, the theory and practice of learning neural networks are undergoing a real "deep revolution", caused by the successful application of Deep Learning methods. Third-generation neural networks, in contrast to the classic second-generation networks of the 80-90s, based on a new learning paradigm, allowed us to get rid of a number of problems that hindered the spread and successful use of traditional neural networks. Networks trained using deep learning algorithms not only exceeded the accuracy of the best alternative approaches, but also showed the beginnings of understanding the meaning of the presented information in a number of tasks (for example, when recognizing images, analyzing text information, and so on). Regression equations (changes in crown height and diameter with age) are obtained and are applicable in models of protective forest stands. It was found that in dry-steppe conditions, with good light and additional moisture, the stages of formation of shoot systems occur more intensively, and a characteristic feature of their development is a reduction in the duration of growth of the main axis and an earlier transition from monopodial to sympodial type of branching of shoots.
The most successfulmodern industrial methods of computer vision and speech recognition are based on the use of deep networks, and the giants of the IT industry buy up teams of researchers engaged in deep neural networks.

Key words : Neural networks, elements off implementation, deep learning, growth, development, crown, models, protective forest stands, chestnut soils.

1. INTRODUCTION

Networks often implementinformation in the form of computer programs, although more and more chips are being produced that implement neural networks by hardware means. The main property of networks is the ability to learn. Deep learning is a set of algorithms that attempt tomodel VAT and high-level abstractions in data using architectures consisting of multiple nonlinear transformations [2-3]. A deep neural network (DNN-Deep Neural Network) is an artificial neural network withseveral hidden layers [1]. Like conventional neural networks, deep neural networks can model complex non-linear relationships between elements. When learning a deep neural network, the resulting model tries to represent an object as a combination of simple primitives (for example, in a facial recognition problem, such primitives can be parts of the face: nose, eyes, mouth, and so on). Additional layers allow you to build abstractions of higher and higher levels, which also allows you to build models for recognizing complex objects in the real world [4].

2. MATERIALS AND METHODS

As a rule, deep networks are built as direct distribution networks. However, recent experiments have shown how deep learning techniques can be applied to recurrent neural networks. Convolutional neural networks are used in the field of machine vision, where this approach has proven to be effective [5]. Convolutional neural networks were also used for speech recognition [6].
Deep neural networks can be trained using a conventional error propagation algorithm. There are a large number of modifications to this algorithm. In this way, several parameters can be used rules for assigning weights. For example, training of weighting coefficients $\omega_{ij}(t)$ by stochastic gradient descent algorithm:

$$\square_{ij}(t + = \square 1) \quad {}_{ij}()t+\square \frac{\partial C}{\ }, \quad (1)$$

where NN is a function for adjusting the value of the current step, *and C* is the loss function. The choice of the loss function can be determined by the class of the machine learning task (with a teacher, without a teacher, with reinforcement) and the activation function [7-10].

The two main problems of deepneural networks are attributed to the same problems that arise in the training of conventional neural networks: training and retraining.

Deep structures are more likely to be retrained, because with more layers to model high-level abstractions, the network can "learn" rare situations. In this case, various types of regularization can help. One possible regularization method (dropout) involves randomly excluding nodes during training. In some cases, this helps you remember less about the riskof addiction in your training data.

Due to the simplicity of implementation and good convergence, the reverse error propagation method and gradient descent are often used for training deep neural networks. However, when learning deep structures, there are several problems that are particularly important when optimizing functions in a large-dimensional space: the number of computational elements, the initial conditions for network weights, and the step constant described above [11].

In addition, the algorithm of stochasticgradient descent is known for its vanishing gradient problem, which consists in weakening the gradient, and therefore the speed of learning as it goes deeper from the last layers of the network to the beginning of the network. Through this, the deep layers of the network learn very poorly. However, recently there is a tendency to use RELU (Rectified Linear Unit) nonlinearity instead of the sigmoid node activation function in deep networks, whose function can be described as *max* (0, *x*). A deep network with this kind of activation function does not have the problem of attenuating the gradient and learns well by gradient descent. In conditions of large dimensions, it is impractical to completely search through all combinations of parameter values.

To speed up calculations, parallelism is used, which is inherent inthe essence of the neural network training algorithm for forward and reverse traversal. Parallelization of the algorithm on T threads is possible at the level:

learning phases with simultaneous training of the network at different settings of its parameters: the number of layers, neurons in layers, cob-x weight settings and an algorithm for controlling the step of their change (T = 2-20);

batch training (T = 10-1000); in this case, the training set is divided into T subsets, its own gradient is calculated for each, the resulting gradients are summed and, thus, the total direction of weight adjustment is obtained;

pipeline training of neural network layers (T = 3-30);

nodes, i.e. neurons of the neural network (T = 100-1000 000 or more);

the weights of the neurons (T = 100 – 10 000 and more);

bit (byte) oriented computing, including stochastic flows, with the appropriate organization of the main processing tools, i.e. adders, multipliers, and memory blocks (T is 1-2 orders of magnitude greater than the values given above);

The last three levels providethe highest parallelism rate and are particularly effective when using hardware to speed up the learning of artificial neural networks, such as GPUs and FPGAs.

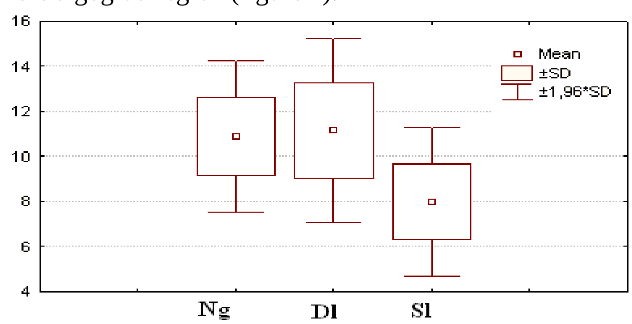
More radical ways to speed up learning include using Extreme Learning Machines, "No prop" neural networks, and leverless neural networks [1].

As we can see from the above, increasing the speed of neural networks is an important and urgent problem. One of the ways to speed up their work by using parallelism, which is inherent in the neural network itself, is its implementation on a chip.

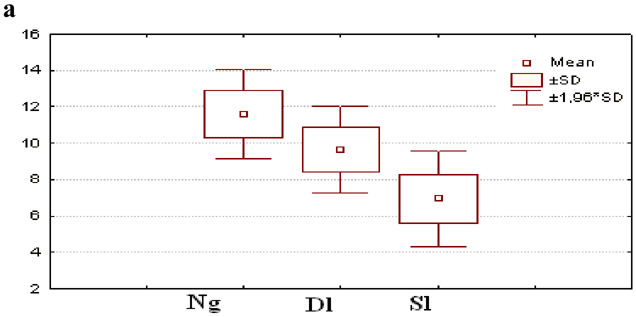
In the literature on creating neural networks on chips, only a few ideas for specific architectures can be traced that can be adapted for modeling neural networks, even less those that have a built-in learning algorithm.

The crystal of the FPGA SPARTAN 3 series has an architecture that allows you to create a neural network for pattern recognition. The WEBPACK XILINX software environment for working with FPGA series chips is free of charge, which provides an advantage in its application [2].

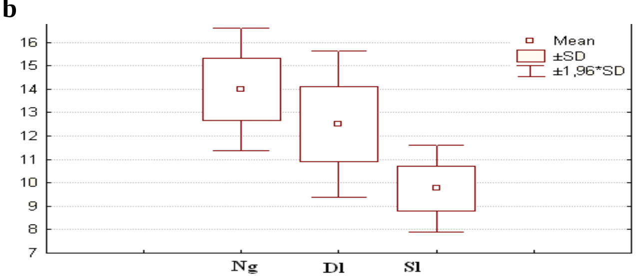
It is revealed that the morphological parameters of leaves determine the development and adaptation to stress factors of the assimilation apparatus. The variability of leaf features allowed us to identify the ecological valence of forms of C. pontica C. Koch, which were introduced for the first time in the Volgograd region (figure 1).



a



b



c

a - 'Circassian-2', b -' Futkurami', c - ' President',

Ng – number of veins, PCs., Dl-leaf length, cm, Sl-leaf width, cm

Figure 1 : variability of leaf morphological features

Work in the Xilinx ISE Webpack programming environment proceeds in the following order:
create a schematic diagram of the projected device in the Xilinx ISE Design Suite 13.2 circuit editor (FPGA Editor) or inthe documentation, with the design of solutions for this device in VHDL or Verilog;
pre-functional (Behavioral Simulation) or temporary modeling to detect errors and check the health of the project being created or its individual parts;
binding the project's outputs tothe crystal's inputs and outputs, selecting source levels, critical contours (Constraints Editor), and so on.;
launch automated placement of the project in the chip and analyze reports that are generated to detect warnings and errors (implementation Design), and in the absence of such and non-critical ones, proceed to the next stage;
project verification, i.e. the final time simulation (Post-Fit Simulation) after placing the project in the chip, with all real delays in signal propagation inside the FPGA chip;
configuring the Creebecame the FPGA with the bitstream (iMPACT 10.1 i). In order to configure the FPGA, you must have a bootable JTAG cable. The bitstream is loaded via dedicated configuration pins using various methods and modes of FPGA loading. After successful loading of the project, the project is checked and adjusted in the future.
If necessary and further development of the design of the project on FPGA all the passed stages are repeated until the complete completion of the project in CILM [12].
Like linear classification and regression methods, neural networks are inherently responsive in the form of:

$$y\,x(\square = 1)\,f\,(\sum_{j=1} \square\,\square_{jj}\,())x \quad (2)$$

where f is the nonlinear activation function, ωis the vector of weights, □ - is the nonlinear basic functions [3].
For linear basic functions, calculations can be implemented with parallel processing of all neurons of the next layer and sequential accumulation of weighted sums of weights for each of them. This solution provides for the use of DSP IP cores included in the latest XILINX FPGA series. This provides flexible management of feature map argument areas, but is quite costly [13].
The option with parallel calculation of the activation function input (post-VNO for each neuron of the next layer of the network) is effectively implemented in more affordable FPGAs of the Spartan3 type. In the block diagram in the figure, THIS option is shown as a pyramid add-on of the values of activation functionsof all neurons of the previous layer weighted with multipliers MUL. The value of activation functions and weighting factors are stored in two-port block memory RAM.

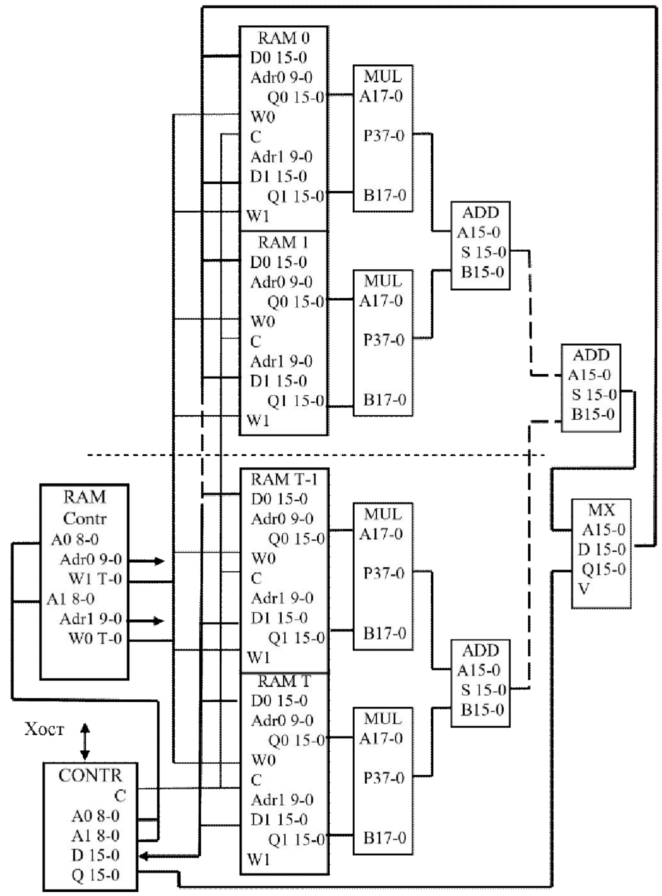


Figure 2: Block diagram of a neural network on an FPGA with parallel calculation of weights

Figure 2 shows us block diagram of neural network.
To work effectively with variable formats of receptive fields and intermediate layers of deep neural networks, an optimal choice of parameter *T*, equal to or multiple of the most commonly used format size, is necessary. In this case, large formats will be processed several times per selectedfile, and smaller formats will be processed by masking the extra inputs with null values at the outputs of the corresponding memory blocks [14].
The CONTR controller provides the necessary sequence of addresses and permissive levels at the write inputs of the corresponding memory blocks. The mosteconomical solution for generating these sequences is to use additional control memory blocks for independent storage and issuing addresses and permissive write levels. The functions of the controller itself then include the formation of significantly simpler cyclich-like address sequences for reading and writing large areas of control memory, as well as initialization of the entire memory of the neural network and downloading the results of calculations [17].
It should be noted that the uniform use of resources distributed in LUT cells for building a pyramid adder on the one hand, and dedicated memory blocks and multipliers on the other, provides a fairly high utilization factor of the FPGA chip area.
The functionality of the Spartan-3 FPGA of various modifications used in the tool module Spartan-3 Starter Board is characterized by the following indicators [15-17]:

there are two types of internal RAM: distributed RAM, implemented on the basis of 4-input conversion tables (LookUp Table, LUT) configure logic blocks, and built-in block memory Block RAM, which can be organized as a two-port RAM;
the amount of internal distributed memory Distributed RAM and built in block memory Block SelectRAM is sufficient for implementing a medium sized neural network;
the use of four digital synchronization control units (DCM), which perform the functions of multiplication, division and phase shifting of clock frequencies, and provide advanced capabilities for controlling clock signals not only inside the chip, but also at the level of the printed circuit Board of the projected system;
high performance, which allows the implementation of projects with system frequencies up to 326 MHz;
the use of a global network of clock signals makes it possible to distribute synchronization signals withina chip with small front differences;
ability to implement fast internal interfaces to external high-performance memory elements (RAM or ROM);
using special accelerated transfer logic for performing high speedarithmetic operations;
built-in hardware multipliers designed to calculate the product of two 18-bit operands;
cascading chains make it possible to implement functions with a large number of input variables;
support for data transmission with double Data Rate (DDR), which opens up wide opportunities for implementing high-speed digital signal processing devices;
the use of Select I/O technology allows you to support 17 single-pole and 6 differential digital signal input / output standards, in particular, LVTTL, LVCMOS12, GTL, SSTL2 (II), HSTL (III), PCI 3.3, AGP, CTT;
full support for the peripheral scanning Protocol in accordance with the IEEE Std 1149.1 (JTAG) and IEEE Std 1532 standards;
support for 5 FPGA configuration modes (Master Serial, Slave Serial, Master Parallel, Slave Parallel, JTAG).

3. CONCLUSION

The prospects of Corylus L. shrubs in light-chestnut soils have been identified, which is associated with indicators of adaptive potential and shoot-forming ability, as well as with compliance with the agrotechnology of creating and operating masterbeds. Cluster analysis showed a correlation of features at a 5% significance level.

Corylus avellana and ‘Circassian-2’ with a pronounced variability of morphological features (18.0-24.4 %), which indicates their broad ecological valence and adaptive capabilities, are recommended for models of protective forest stands in chestnut soils. Varieties of Corylus pontica also showed good growth and development of architectonics in light-chestnut soils, where they reach the same heights as in the natural range of growth. The best results (the shape and density of the crown, the nature of the foliage) were noted in the form of 'Circassian-2'.

Species of the genus Corylus L. are valuable shrubs for protective forest stands (for fixing slopes, ravines and slopes). They form the forest floor, which is a source of organic matter and a factor in the flow-regulating properties of plantings. The forest floor has a high water capacity and creates roughness for runoff, which provides favorable conditions for the transfer of surface runoff to the intra-soil. The spatial influence of plantings depends on their purpose, location, and soil and climate conditions. Corylus avellana is included in the list of recommended species for green construction, cultivated as ornamental plants.

Thus, the development and construction of neural networks in FPGA allows you to significantly speed up the processes of their functioning and learning by using the possibility of parallel processing from hundreds to several thousand computational threads. The proposed variants of the neural network structure are characterized by a high coefficient of crystal utilization and the possibility of application for the implementation ofneural networks.

The research was carried out on the topic of state task №0713-2019-0004 of the Federal Scientific Center for Agroecology, Complex Melioration and Protective Afforestation of the Russian Academy of Sciences.

REFERENCES

1. Dolgih A.A. **Monitoring of introduction resources of the Kulunda arboretum and allocation of valuable gene pool for protective afforestation**. World Ecology Journal. 2018. Vol. 8 No. 1 p. 29-42 DOI: 10.25726/NM.2018.1.1.003
2. Germain E. **The reproduction of hazelnut** (Corylus avellana L.). Acta Hortic. 1994. 351. Pp. 195-210. DOI: 10.17660/ActaHortic.1994.351.19
3. Kalmykova A.L., Tereshkin A.V., Zaigralova G.N., Eskov D.V., Kornienko M.Yu. **Analysis of the range of trees and shrubs used in the landscaping of Saratov**. Amazonia Investiga. 2019. T. 8. №22. Pp. 510-521.
4. Kruzhilin S.N., Taran S.S., Semenyutina A.V., MatvienkoE.Yu. **Growth peculiarities and age dynamics of Quercus robur L. formation in steppe region conditions**. Kuwait Journal of Science. 2018. Vol. 45. Issue 4. P. 52-58
5. Kulik K.N., Barabanov A.T., Manaenkov A.S., Kulik A.K. **Forecast assumption and analysis of the development of protective afforestation in the Volgograd region**. Studies on Russian Economic Development. 2017. T. 28. № 6. pp. 641-647.
6. Larionov M.V., Larionov N.V., Siraeva I.S., Ermolenko A.S. **The Composition and Characteristics of the Dendroflora in the Transformed Conditions of the Middle Reaches of the River Khoper)**. IOP Conference Series: Earth and Environmental Science. 2018. Vol. 115, Issue 1. № 012009 DOI: 10.1051/cagri/2018011
7. Leadbetter C.W., Capik J.M., Pisetta M., Molnar T.J. **Sources of resistance to eastern filbert blight in hazelnuts from the Republic of Georgia**. Scientia Horticulturae. 2015. 193. Pp. 269-275. DOI: 10/1016/j.scienta.2015.07.017

8. Lombardoni J.J., Honig J.A., Vaiciunas J., Kubik C., Capik J., Mehlenbacher S., Molnar T.J. **Evaluation of European hazelnut (Corylus avellana) genetic diversity using a genotyping-by-sequencing approach**. Acta Horticulturae. 2020. Vol. 1280. Pp. 11-16. DOI: 10.17660/ActaHortic.2020.1280.2
9. Melikhov V.V., Novikov A.A., Medvedeva L.N., Komarova O.P. **Green technologies: the basis for integration and clustering of subjects at the regional level of economy**. Contributions to Economics. 2017. №9783319454610. С. 365-382.
10. Molnar T.J., Goffreda J.C., Funk C.R. **Developing hazelnuts for the eastern Unites States**. Acta Horticulturae. 2005. 686. Pp. 609-618. DOI: 10/1766/ActaHortic.2005.686.81.
11. Molnar T.J., Zaurov D.E., Goffreda J.C., Mehlenbacher S.A. **Survey of hazelnut germplasm from Russia and Crimea for response to eastern flibert blight**. HortScience. 2007. 42(1). Pp. 51-56. DOI: 10.21273/hortsci.42.1.51
12. Semenyutina A.V., Kostyukov S.M. **Bioecological justification assortment of shrubs for landscaping urban landscapes**. – Montreal, Accent Graphics Communications, 2013. – 164 p.
13. Semenyutina A.V., Podkovyrov I.Y., HuzhahmetovaA.Sh., Semenyutina V.A., Podkovyrova G.V. **Mathematical justification of the selection of wood plants biodiversity in the reconstruction of objects of gardening**. International Journal of Pure and Applied Mathematics. 2016. Vol. 110. №2. p. 361-368.
14. Semenyutina A.V., Podkovyrova G.V., KhuzhahmetovaA.Sh., Svintsov I.P., Semenyutina V.A., PodkovyrovI.Yu. **Engineering implementation of landscaping of low-forest regions**. International journal of mechanical engineering and technology. 2018. Vol.9. Issue 10. pp. 1415-1422.
15. Volk G., Samarina L., Kulyan R., Gorshkov V., Malyarovskaya V., Ryndin A., Polek M.L., Krueger R., Stover E. **Citrus genebank collections: international collaboration opportunities between the US and Russia** // Genetic resources and crop evolution. 2018. T. 65. №2. P. 433-447. DOI:10.1007/s10722-017-0543-z.
16. Ahmmad, S. N. Z., Jaafar, M. M., Muchtar, F., & Yusof, M. I. (2020). **Automated gardening portable plant using IoT.** International Journal of Advanced Trends in Computer Science and Engineering, 9(1.1 Special Issue). https://doi.org/10.30534/ijatcse/2020/3691.12020
17. Ahmmad, S. N. Z., &Muchtar, F. (2020). **A review on applications of optimization using bat algorithm**. International Journal of Advanced Trends in Computer Science and Engineering, 9(1.1 Special Issue), 212–219. https://doi.org/10.30534/ijatcse/2020/3791.12020