Neural Network for Low-Memory IoT Devices and MNIST Image Recognition Using Kernels Based on Logistic Map

Andrei Velichko

Abstract—The study presents a neural network, which uses filters based on logistic mapping (LogNNet). LogNNet has a feedforward network structure, but possesses the properties of reservoir networks. The input weight matrix, set by a recurrent logistic mapping, forms the kernels that transform the input space to the higher-dimensional feature space. The most effective MNIST handwritten digit recognition occurs under chaotic behavior of logistic map. An advantage of LogNNet implementation on IoT Devices is the significant savings in used memory (more than 10 times) compared to other neural networks. The presented network architecture uses an array of weights with a total memory size from 1 kB to 29 kB and achieves a classification accuracy of 80.3–96.3%.

Index Terms—handwritten digits recognition, Mnist, neural network, IoT, reservoir computing, logistic map.

I. INTRODUCTION

In the age of neural networks and Internet of Things (IoT) developments, the search for new neural network architectures capable of operating on devices with a small amount of memory (tens of kB RAM) becomes an urgent agenda [1], [2]. The ability to perform processing of incoming information on the spot, without sending it to the cloud, is an important feature of intelligent IoT devices. Currently, effective memory redistribution algorithms in convolutional neural networks (CNN) are available and do not exceed 2kB of consumed RAM [3]. However, the complexity of the algorithms leads to the large size of the program itself. An alternative approach to reduce the number of trained weights is based on physical reservoir computing (RC) [4]. RC uses complex physical dynamic systems (coupled oscillators [5]–[7], memristor crossbar arrays [8], opto-electronic feedback loop [9]), or recurrent neural networks (echo state networks (ESNs)) [10] and liquid state machines (LSMs) [11]), as reservoirs with rich dynamics and powerful computing capabilities. The couplings in the reservoir are not trained, but are specified in a special way. The reservoir translates the input data into a higher dimensionality of space (kernel trick [12]), and the output neural network, after training, can classify the result more correctly. The search for simple algorithms for simulating complex reservoir dynamics is an important research task. The current paper presents a new architecture for a neural network, where complex dynamics is simulated by the application of a logistic mapping to the weights of a multilayer feedforward network. Previously, logistic mapping was used in neural networks as activation function [13], [14] and as a model object [15], [16].

II. DATA AND METHODS

The database of handwritten numbers MNIST was used for the study, consisting of 70,000 images of handwritten digits from “0” to “9”. Each image is 28 × 28 pixels in size (see Fig. 1a). The database is divided into two sets. The first set consists of 60,000 images intended for training the network, and the second set of 10,000 images is used to test the network and to calculate classification accuracy. The image is presented in grayscale with the intensity of each pixel in the range from 0 to 255. Before inputting the network, a two-dimensional image was converted into a one-dimensional array $Y[i]$, where $i = 1..784$, using special transformation T-patterns. The main types of T-patterns used are presented in Fig. 1b, c, d. In addition, T-patterns can be used for reverse transformation from a one-dimensional array to a two-dimensional array.

Fig. 1. An example of a 28 × 28 pixel image of the handwritten digit “8” from the MNIST database (a). Patterns for transforming the image array: T-pattern-1 (b), T-pattern-2 (c), T-pattern-3 (d).

T-pattern-1 performs a column-by-column input of the image, T-pattern-2 converts the image in a spiral (clockwise) form, and T-pattern-3 is a combination of line-by-line input of the central part of the image, with the addition of borders in a spiral (clockwise) form.

After the T-pattern transformation, the values of the array $Y$ are normalized by dividing each element of the array by 255, and a bias element $Y[0] = 1$ is added.

A. Velichko is with the Institute of Physics and Technology, Petrozavodsk State University, 31 Lenina str., Petrozavodsk 185910, Russia (e-mail: velichko@petrsu.ru).

Manuscript received ********. This work was supported by the Russian Science Foundation (grant no. 16-19-00135).
A. Network architecture

The network architecture used in the current study is presented in Fig. 2a. Data processing is performed similarly to the feedforward network (Fig. 2b), according to the equations

\[ S_h = f_h(Y \cdot W_1), \quad S_{\text{out}} = f_{\text{out}}(S_h \cdot W_2), \]

where \( W_1, W_2 \) are the weight matrices, \( S_h \) is the hidden layer with the number of active elements \( P \) and the bias element \( S[0][n] = 1 \), \( S_{\text{out}} \) is the output layer, \( f_h \) is identity activation function, which is normalized in the range from -0.5 to 0.5, and \( f_{\text{out}} \) is logistic activation function.

The weights \( W_1 \) in the reservoir are set recursively, based on the logistic mapping

\[ x_{p+1} = 1 - r \cdot x_p^2, \]

where \( r \) is a positive parameter. Applying (2) to the elements of the weight matrix \( W_1 \), the following equation is obtained

\[ W_1[i][p+1] = 1 - r \cdot (W_1[i][p])^2. \]

The initial values of the elements of the first row \( W_1[i][1] \) are set by the equation

\[ W_{\text{in}}[i][1] = A \cdot \sin\left(\frac{i}{784} \cdot \frac{\pi}{B}\right), \]

where \( A \) and \( B \) are adjustable parameters, \( A = 0.3, B = 5.9, i = 0..784 \).

The classification of handwritten digits in the range “0” - “9” takes place according to the largest value of the element of the output layer \( S_{\text{out}}[n] \), where \( n = 0..9 \). The training of the weights \( W_2 \) is performed by the error back-propagation method [17] with the learning rate of 0.3. The initial values of the weights \( W_2 \) are set randomly between -0.5 and 0.5.

Weights \( W_1 \) are not trained, and only the weights \( W_2 \) of the output classifier are trained. This approach resembles the operation of a recurrent reservoir network, except the recurrence transformation is applied not to the input data, but to the elements of the matrix \( W_1 \).

The logistic mapping is widely known as the simplest model that demonstrates the transition to chaotic behavior through a sequence of period doubling bifurcations (Feigenbaum’s script [18]). The cascade of period doubling bifurcations in the logistic mapping can be visualized using a phase-parameter diagram, which is also called a one-parameter bifurcation diagram (Fig. 3).

Fig. 3. The bifurcation diagram of the logistic mapping (2).

In the text, brief information on the number of network layers and neurons is indicated in the format LogNNet-784:P:10.

III. Result and Discussion

The dependencies of classification accuracy on the number of training epochs for different T-patterns are shown in Fig. 4. As T-pattern-3 provides the best results, all subsequent calculations are given with this transformation. The application of transformations to the initial images should be considered as a heuristic technique, which allows to overcome the disadvantages of back-propagation of errors learning method. Configuring a neural network is truly considered an art [19], and many authors use affine transformations [20], [21] and elastic deformations [22].

Fig. 4. The dependence of the classification accuracy on the number of epochs, for different T-patterns. The number of reservoir neurons is \( P = 25 \) (a), \( P = 100 \) (b) and \( r = 1.885 \).

The dependence of the classification accuracy on the parameter \( r \) is presented in Fig. 5. The accuracy grows with increasing \( r \) and increasing number of neurons \( P \). At values of \( r = 1.65, \ 1.805, \ 1.885, \ 1.967, \ 1.992 \) local maximums of classification accuracy is observed. The shape of the graph resembles the dependence of the Lyapunov exponent on \( r \) [18], and indicates the importance of the chaotic behavior of the logistic mapping in the recognition process. The important role of the chaotic dynamics of the reservoir was highlighted in the studies on ESNs (setting the spectral radius) [10], [23], and LSMs (setting separation property) [11].

The shape of distribution of weights \( W_1 \) for different reservoir neurons, at \( r = 1.885 \), is presented in Fig. 6 (for convenience of visualization, the inverse transformation T-pattern-1 was applied, and the element \( W_1[0][p] \) was placed in the upper left corner). Due to the formation of matrix elements through the logistic relation (2), the kernels shape changes with each step \( p \). Kernels transform the input space to the higher-dimensional feature space. The result of the kernels operation is released by the reservoir neurons and transmitted to the output classifier.

Fig. 5. The dependence of the classification accuracy on the parameter \( r \) for different T-patterns. The number of reservoir neurons is \( P = 25 \) (a), \( P = 100 \) (b).
Table 1 gives comparative estimates of the classification accuracy and the amount of allocated memory for the weight arrays for various network configurations, including for LogNNet with 1-2 layer output classifier (Alg.-2, \(r = 1.885\)), and for 1-2 layer feedforward networks Fig. 2.b.c. Applying Alg.-1, \(MeW\) values can be reduced by 3 kB, and reach \(MeW \sim 1\) kB for LogNNet-784:25:10, although Alg.-1 is slower than Alg.-2.

Table 2 summarises a comparison of the processing times of a single image by the LogNNet network on the IoT device Raspberry Pi 4 for a different number of neurons in reservoir \(P\). The execution times (\(t_{\text{alg2}}\) and \(t_{\text{alg3}}\)) reflect the application of the one-dimensional array \(W_i\) (Alg.-2) and the two-dimensional array \(W_i\) (Alg.-3), respectively.

**IV. CONCLUSION**

The study presents the neural network LogNNet, which uses filters based on logistic mapping to achieve high accuracy indicators for the classification of handwritten digits from the MNIST database. The network uses less memory for weight arrays than many well-known network configurations. The future research directions may lie in the search for other chaotic mappings and initial conditions (3) for more efficient operation of LogNNet.
REFERENCES

[1] S. Li, Y. Dou, J. Xu, Q. Wang, and X. Niu, “mmCNN: A Novel Method for Large Convolutional Neural Network on Memory-Limited Devices,” in $2018$ IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), 2018, vol. 1, pp. 881–886, doi: 10.1109/COMPSAC.2018.00152.

[2] S. Abdel Magid, F. Petrini, and B. Derfoul, “Image classification on IoT edge devices: profiling and modeling,” Cluster Comput., 2019, doi: 10.1007/s10586-019-02971-9.

[3] A. Gural and B. Murmann, “Memory-Optimal Direct Convolutions for Maximizing Classification Accuracy in Embedded Applications,” in Proceedings of the 36th International Conference on Machine Learning, 2019, vol. 97, pp. 2515–2524, [Online]. Available: http://proceedings.mlr.press/v97/gural19a.html.

[4] G. Tanaka et al., “Recent advances in physical reservoir computing: A review,” Neural Networks, vol. 115, pp. 100–123, 2019, doi: https://doi.org/10.1016/j.neunet.2019.03.005.

[5] A. A. Velichko, D. V Ryabokon, S. D. Khanin, A. V Sidorenko, and A. G. Rikkiev, “Reservoir computing using high order synchronization of coupled oscillators,” IOP Conf. Ser. Mater. Sci. Eng., vol. 862, p. 52062, 2020, doi: 10.1088/1757-899x/862/5/052062.

[6] T. Yamane, Y. Katayama, R. Nakane, G. Tanaka, and D. Nakano, “Wave-Based Reservoir Computing by Synchronization of Coupled Oscillators BT - Neural Information Processing,” 2015, pp. 198–205.

[7] Velichko, A. A. Method for Evaluating Chimeric Synchronization of Coupled Oscillators and Its Application for Creating a Neural Network Information Converter,” Electronics, vol. 8, no. 7, p. 756, Jul. 2019, doi: 10.3390/electronics8070756.

[8] C. Donahue et al., “Design and analysis of neurmemristive echo state networks with limited-precision synapses,” in 2015 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), 2015, pp. 1–6, doi: 10.1109/CISDA.2015.7208623.

[9] L. Larger, A. Baylón-Fuentes, R. Martinenghi, V. S. Udaltsov, Y. K. Chembo, and M. Jacquot, “High-Speed Photonic Reservoir Computing Using a Time-Delay-Based Architecture: Million Words per Second Classification,” Phys. Rev. X, vol. 7, no. 1, p. 11015, Feb. 2017, doi: 10.1103/PhysRevX.7.011015.

[10] M. C. Ozturk, D. Xu, and J. C. Principe, “Analysis and Design of Echo State Networks,” Neural Comput., vol. 19, no. 1, pp. 111–138, Nov. 2006, doi: 10.1162/neco.2007.19.1.111.

[11] P. Wijesinghe, G. Srinivasan, P. Panda, and K. Roy, “Analysis of Liquid Ensembles for Enhancing the Performance and Accuracy of Liquid State Machines,” Front. Neurosci., vol. 13, p. 504, 2019, doi: 10.3389/fnins.2019.00504.

[12] M. Lukoševičius and H. Jaeger, “Reservoir computing approaches to recurrent neural network training,” Comput. Sci. Rev., vol. 3, no. 3, pp. 127–149, Aug. 2009, doi: 10.1016/j.cosrev.2009.03.005.

[13] M. Azarpour, S. A. Seyedsalehi, and A. Taherkhani, “Robust pattern recognition using chaotic dynamics in Attractor Recurrent Neural Network,” in The 2010 International Joint Conference on Neural Networks (IJCNN), 2010, pp. 1–6, doi: 10.1109/IJCNN.2010.5596375.

[14] T. Wang and N. Jia, “A GCM neural network using cubic logistic map for information processing,” Neural Comput. Appl., vol. 28, no. 7, pp. 1891–1903, 2017, doi: 10.1007/s00521-016-2407-4.

[15] J. P. L. Tan, “Simulating extrapolated dynamics with parameterization networks,” Feb. 2019, Accessed: Jun. 01, 2020. [Online]. Available: http://arxiv.org/abs/1902.03440.

[16] A. Margaris, E. Kotsialos, N. Kofidis, M. Roumeliotis, and M. Adamopoulos, “Logistic map neural modelling: A theoretical foundation,” Int. J. Comput. Math., vol. 82, no. 9, pp. 1055–1072, Sep. 2005, doi: 10.1080/00207160500113033.

[17] R. Callan, Essence of Neural Networks. USA: Prentice Hall PTR, 1998.

[18] B. Luque, L. Lacasa, F. J. Ballesteros, and A. Robledo, “Feigenbaum Graphs: A Complex Network Perspective of Chaos,” PLoS One, vol. 6, no. 9, p. e22411, Sep. 2011, [Online]. Available: https://doi.org/10.1371/journal.pone.0022411.

[19] G. Montavon, G. B. Orr, and K.-R. Müller, Eds., “Neural Networks: Tricks of the Trade - Second Edition,” vol. 7700. Springer, 2012, doi: 10.1007/978-3-642-35289-8.

[20] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, 1998, doi: 10.1109/5.726791.

[21] N. Schaeetti, M. Salomon, and R. Couturier, “Echo State Networks-Based Reservoir Computing for MNIST Handwritten Digits Recognition,” in 2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES), 2016, pp. 484–491, doi: 10.1109/CSE-EUC-DCABES.2016.229.

[22] P. Y. Simard, D. Steinkraus, and J. C. Platt, “Best practices for convolutional neural networks applied to visual document analysis,” in Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings., 2003, pp. 958–963, doi: 10.1109/ICDAR.2003.1227801.

[23] S. Krishnagopal, Y. Aloimonos, and M. Girvan, “Similarity Learning and Generalization with Limited Data: A Reservoir Computing Approach,” Complexity, vol. 2018, p. 6953836, 2018, doi: 10.1155/2018/6953836.