

Балабанов Дмитрий Валерьевич – Южный федеральный университет; e-mail: dbalabanov@sfedu.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 88634371651; кафедра систем автоматизированного проектирования; аспирант.

Ковтун Антон Владиславович – e-mail: anton.kovtun93@gmail.com; кафедра систем автоматизированного проектирования; аспирант.

Кравченко Юрий Алексеевич – e-mail: yakravchenko@sfedu.ru; кафедра систем автоматизированного проектирования; доцент.

Balabanov Dmitry Valerievich – Southern Federal University; e-mail: dbalabanov@sfedu.ru; 44, Nekrasovskiy lane, Taganrog, 347928, Russia; phone: +78634371651; the department of computer aided design; graduate student.

Kovtun Anton Vladislavovich – e-mail: anton.kovtun93@gmail.com; the department of computer aided design; graduate student.

Kravchenko Yury Alekseevich – e-mail: yakravchenko@sfedu.ru; the department of computer aided design; associate professor.

УДК 004.032.26

DOI 10.18522/2311-3103-2020-3-146-156

В.В. Бахчевников, В.А. Деркачев, А.Н. Бакуменко

СПОСОБ ИСПОЛЬЗОВАНИЯ СРЕДСТВ БЫСТРОГО ПРОТОТИПИРОВАНИЯ ДЛЯ РЕАЛИЗАЦИИ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ НА ПЛИС

Исследования в области искусственного интеллекта ведутся с возрастающим интересом с каждым годом. Области применения искусственного интеллекта довольно обширны: автоматизация, анализ большого объема данных, технологии умного дома, машинное зрение и т.д. Технологии искусственного интеллекта базируются на использовании искусственных нейронных сетей, имеющие в своей основе принципы нервной системы животных. При этом актуальным вопросом является реализация искусственных нейронных сетей на различных программно-аппаратных платформах: программируемых логических интегральных схемах (ПЛИС) типа FPGA (Field Programmable Gate Array), на интегральных схемах специального назначения (Application-Specific Integrated Circuit, ASIC), GPU, CPU и т.д. ПЛИС наилучшим образом проявляют себя в малоомощных мобильных системах. ASIC демонстрируют наибольшую производительность с недостатком: высокая цена разработки. Проблема быстрого прототипирования проектов, основанных на использовании искусственных нейронных сетей, для ПЛИС привычными методами (с помощью HDL-языков, HDL-кодеров, графического программирования) заключается в том, что либо такой проект сложен и длителен в отладке (HDL-языки), либо не оптимален получающийся код (HDL-кодеры), либо высока длительность разработки проекта и сложность реконфигурации нейронной сети (графическое программирование). Поэтому в рамках данной работы рассматривается эффективный метод проектирования полносвязных и сверточных нейронных сетей для их реализации на ПЛИС использованием пакета Xilinx System Generator for DSP и Matlab/Simulink. Генерируемые таким образом искусственные нейросети легко реконфигурируемы и позволяют решать следующие задачи: распознавание изображений, оптимальная фильтрация (например, для задач подповерхностной радиолокации).

Искусственный интеллект; искусственные нейронные сети; реализация на ПЛИС; сверточная нейросеть; метод проектирования.

V.V. Bakhchevnikov, V.A. Derkachev, A.N. Bakumenko

USING FAST PROTOTYPING FACILITIES FOR IMPLEMENTATION OF A CONVOLUTION NEURAL NETWORK ON A FPGA

Research in the field of artificial intelligence is carried out with increasing interest every year. The fields of application of artificial intelligence are quite extensive: automation, analysis of a large amount of data, smart home technology, machine vision, etc. Artificial intelligence technologies are based on the use of artificial neural networks, which are based on the principles of the animal nervous system. In this case, the actual issue is the implementation of artificial neural networks on various software and hardware platforms: programmable logic integrated circuits of the FPGA type (Field Programmable Gate Array), on special purpose integrated circuits (Application-Specific Integrated Circuit, ASIC), GPU, CPU etc. FPGA performs best in low-power mobile systems. ASIC demonstrates the highest performance at a fairly high development cost. The problem of rapid prototyping of projects based on the use of artificial neural networks for FPGAs using conventional methods (using HDL languages, HDL encoders, graphic programming) is that either such a project is complex and time-consuming to debug (HDL languages), or the resulting code is not optimal (HDL encoders), or the duration of the project development and the complexity of reconfiguring the neural network (graphical programming) are high. Therefore, in the framework of this work, an effective method for designing fully connected and convolutional neural networks for their implementation on FPGAs using the Xilinx System Generator for DSP and Matlab / Simulink package is considered. Artificial neural networks generated in this way are easily reconfigurable and allow solving the following problems: image recognition, optimal filtering (for example, for problems of subsurface radar).

Artificial Intelligence; artificial neural networks; FPGA implementation; convolutional neural network; design method.

Введение. За последнее время значительно возрос интерес к изучению и применению искусственных нейросетей. В данной работе уделяется особое внимание разработке искусственных нейронных сетей (ИНС) на ПЛИС, как платформе, наилучшим образом проявляющей себя в энергоэффективных решениях [1, 2]. Существует проблема разработки проектов для ПЛИС привычными методами, основанных на использовании ИНС, как полносвязных (ПНС), так и сверточных (СНС). Заключается она в том, что либо такой проект сложен и длителен в отладке (HDL-языки) [3–5], либо не оптимален с точки зрения получаемого кода (HDL-кодеры) [6]. В работах [7–9] описана разработка ПНС и СНС для ПЛИС с помощью Xilinx System Generator for DSP, однако проектирование ПНС с большим количеством нейронов представляет немалые временные сложности. Для устранения данной проблемы и предлагается метод, описанный в данной работе. Метод заключается в проектировании СНС с использованием библиотеки Xilinx System Generator for DSP, импортированную в среду Matlab/Simulink, и использовании встроенного языка MatlabC для автоматической генерации заданной структуры ИНС. Генерируемые по данному методу искусственные нейросети легко реконфигурируемы и позволяют решать следующие задачи: распознавание изображений, оптимальная фильтрация (например, для задач подповерхностной радиолокации [10]). Очевидно, что применение методов искусственного интеллекта на основе нейросетей позволит, в том числе, вывести на качественно новый уровень процедуры распознавания технических состояний при контроле и диагностике сложных объектов [11], а также стать практической альтернативой строгим математическим методам [12] по решению задач идентификации объектов различных типов.

Целью работы является разработка и верификация метода проектирования сверточной нейронной сети для прототипирования на ПЛИС с помощью средств Xilinx System Generator for DSP и Matlab.

Разрабатываемый метод. На рис. 1 представлен полный маршрут проектирования ИНС на ПЛИС.

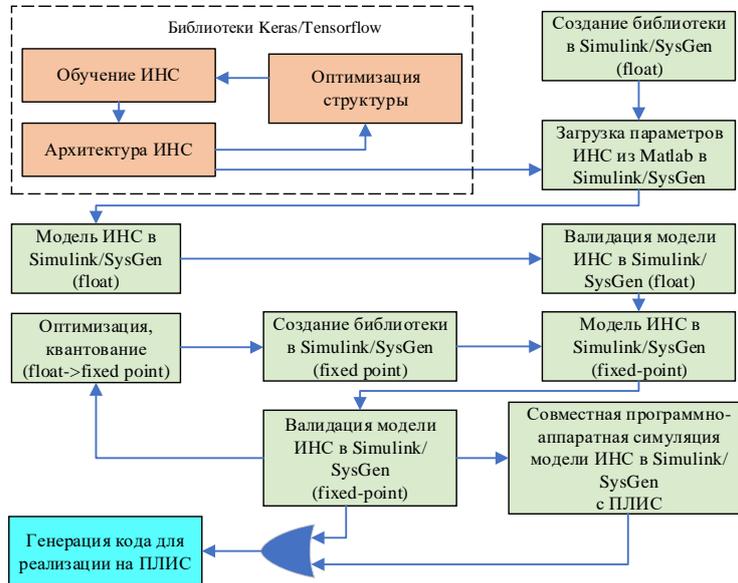


Рис. 1. Маршрут разработки ИНС на ПЛИС для предложенного метода

На представленной схеме оранжевым цветом выделены те процедуры, которые производятся с помощью инструментов проектирования, обучения и тестирования ИНС с использованием соответствующих библиотек (TensorFlow, Keras). Зеленым цветом выделены операции, производимые в Simulink/Xilinx SysGen и необходимые для подготовки проекта к реализации ИНС на целевую платформу с ПЛИС. Библиотека Xilinx элементов описана в [13]. Создание собственной библиотеки необходимо для возможности осуществления автоматической генерации архитектуры ИНС в Simulink/SysGen. Описание работы с Simulink и генерации модели с помощью команд представлено в [14].

Рассмотрим структуру ИНС, которую будем брать для автоматической генерации по разрабатываемому методу. Обобщенная структура одноканальной сверточной нейросети представлена на рис. 2.

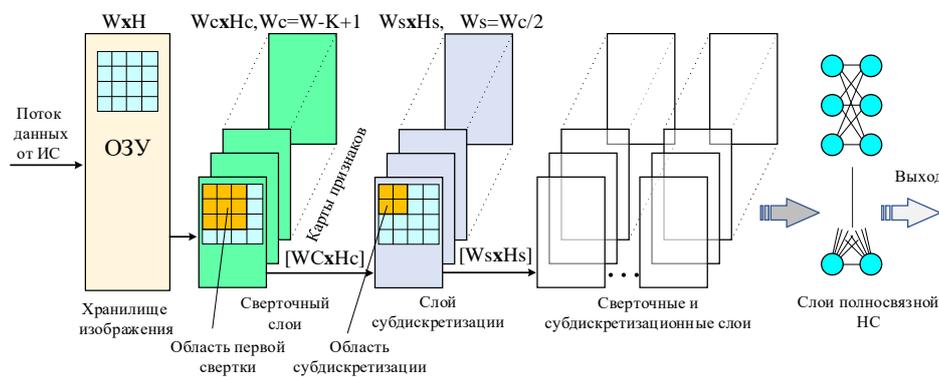


Рис. 2. Структура сверточной нейросети (1 канал)

Здесь блок «Хранилище изображения» представляет собой набор компонентов (RAM, ROM), позволяющих хранить в ячейках RAM изображение, подаваемое на вход в потоковом виде.

В задачи сверточного слоя входит формирование двумерной свертки между входным изображением и ядром фильтра свертки [2, 5,15]. Выходной сигнал сверточного слоя представлен выражением (1).

$$f_i^{CLout} = F_{act} \left(\sum_{j=1}^{n_m} f_j^{in} * k_{i,j} + b_i \right), 1 \leq i \leq n_{out}, \quad (1)$$

где $k_{i,j}$ – фильтры свертки, b_i – значения смещения, F_{act} – функция активации [2]. Методы реализации двумерной свертки на ПЛИС описаны в [16].

Слой субдискретизации производит сокращение изображения в 2 раза (по горизонтали и вертикали) [Ошибка! Источник ссылки не найден.]. Сигнал на выходе слоя субдискретизации (MaxPooling) основан на применении формулы (2).

$$f_{i,j}^{SDout} = \max_{p \times p} (f_{m,n}^{in}, f_{m,n+1}^{in}, \dots, f_{m,n+p-1}^{in}, f_{m+1,n}^{in}, f_{m+1,n+1}^{in}, \dots, f_{m+1,n+p-1}^{in}, \dots, f_{m+p-1,n}^{in}, f_{m+p-1,n+1}^{in}, \dots, f_{m+p-1,n+p-1}^{in}) \quad (2)$$

где f^{in} – значения пикселей входного изображения.

После осуществления операций двумерной свертки и субдискретизации сигнал со всех потоков, связанных с количеством карт признаков, мультиплексируется в один поток и подается на полносвязную часть ИНС [15]. Уравнением (3) описывается расчет сигнала на выходе i -го нейрона полносвязной НС [2].

$$f_i^{FNout} = F_{act} \left(\sum_{j=1}^{n_m} f_j^{in} \cdot w_j + b_i \right), \quad (3)$$

где w_j – весовые коэффициенты, b_i – значения смещения, F_{act} – функция активации.

Функция активации F_{act} применяется после операции свертки, а также в нейронах ПНС [15]. Для СНС в качестве ФА чаще всего используют RELU, в нейронах выходного слоя ПНС – гиперболический тангенс TanH. Также на практике применяются и другие ФА [17].

Автоматическая генерация структуры нейросети предполагает написание скриптов в Matlab для каждого отдельного блока Simulink/SysGen (слои свертки, субдискретизации, нейронов ПНС), а также их соединения между собой в единую структуру.

Генерация сверточной нейросети. В качестве примера будем рассматривать СНС со структурой, представленной в табл. 1.

Таблица 1

Пример анализируемой СНС

Слой НС	Input	Conv2D	MaxPool2D	Flatten	Dense_1	Dense_2
Размерность сигнала	6x6	4x4x3 (ядро 3x3)	2x2x3	1x12	12x32	32x2

Блок «Хранилище изображения» на блоках Simulink/SysGen в сгенерированном для примера из табл. 1 виде представлен на рис. 3. Данный блок является уникальным для каждого канала (цвет) и общим для всех карт признаков следующего слоя. Для ускорения производительности в схему включена двухпортовая ОЗУ, чтобы производить чтение одновременно с записью. Чтение производится после записи первых двух строк изображения плюс количество пикселей, равное линей-

ному размеру сверточного ядра. В ПЗУ записываются массивы вспомогательных значений адреса так, чтобы данные в ОЗУ были представлены в «квадратичной» форме. Это нужно, чтобы как можно скорее производилась операция субдискретизации, работающая с квадратами пикселей 2x2.

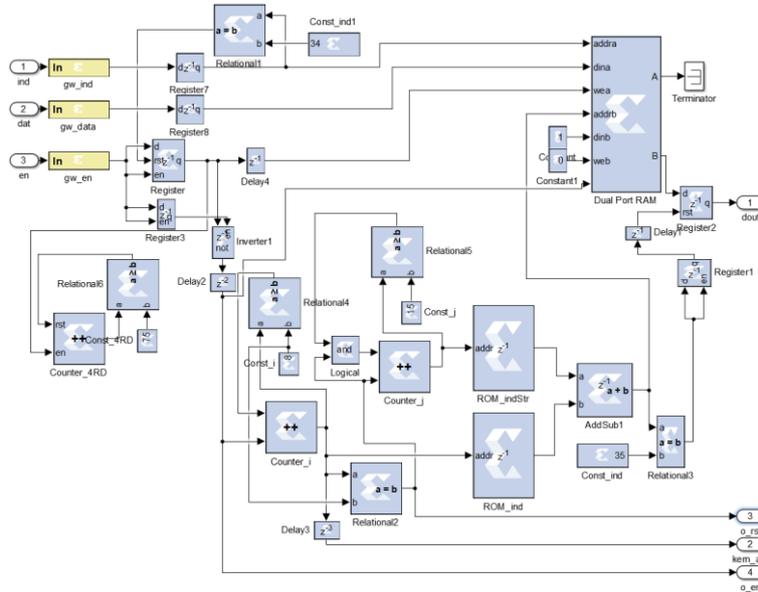


Рис. 3. Блок хранения изображения

На рис. 4,а представлен блок двумерной свертки, производящей операцию из выражения (1). Функциональная схема свертки основана на умножении с накоплением. Рис. 4,б представляет собой функциональную схему слоя субдискретизации для одной карты признаков, работа которой описывается формулой (2). Режим данной схемы является потоковым, поэтому необходимо в блоке хранения изображения использовать квадратичную форму представления информации.

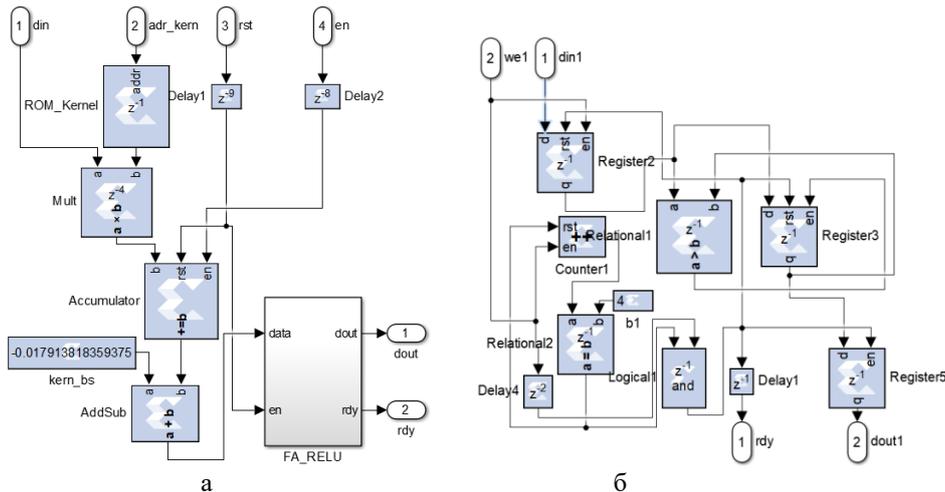


Рис. 4. Блоки для одной карты признаков: а – двумерной свертки, б – субдискретизации

Блок мультиплексирования данных со всех карт признаков в один поток представлен на рис. 5. Данный блок необходим для дальнейшей обработки сигнала в ПНС в потоковом режиме.

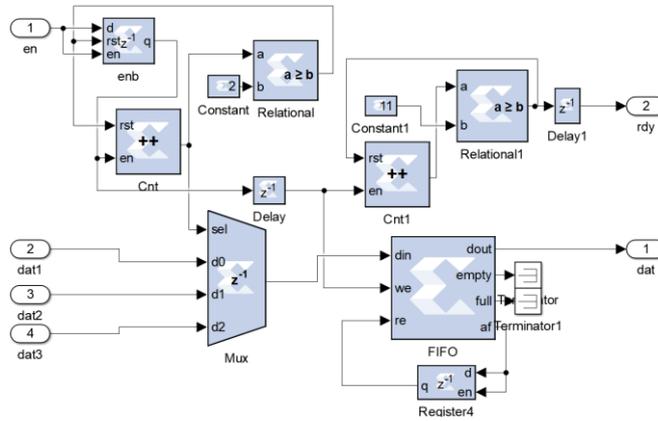


Рис. 5. Блок мультиплексирования в общий поток (Flatten)

На рис. 6,а изображен функциональный блок управления и подготовки сигнала для нейронов первого слоя ПНС. Для последующих слоев ПНС данный блок будет еще содержать мультиплексор, так данные перед этим блоком представляются в параллельном виде. Нейроны ПНС, чья функциональность описывается формулой (3), построены по схеме умножения с накоплением, представленной в работе [8], и последующим применением функции активации, либо RELU, либо TanH (рис. 6,б).

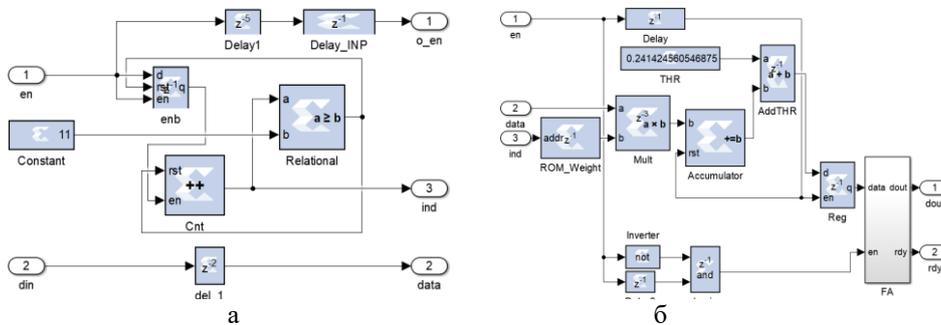


Рис. 6. Блоки: а – управления слоя ПНС, б – нейрон ПНС

Автоматическая генерация СНС позволяет на основе представленных выше функциональных блоков строить нейросеть любой конфигурации, подставляя необходимые значения в некоторые элементы блоков (константы, ПЗУ и т.д.).

Результаты работы сверточной нейросети. Проведем анализ работы СНС, построенной по архитектуре из табл. 1. Обучение нейросети проводилось с помощью средств Keras/TensorFlow. Коэффициенты фильтров свертки и весовые коэффициенты для полносвязной НС загружались в Matlab из файла *.h5. Классификация производилась по двум классам изображений: изображение нуля и единицы.

В целях уменьшения потребляемых ресурсов для функции активации «гиперболический тангенс» была использована его аппроксимация, описанная в работе [18] и представлена выражением (4).

$$FA_{TANH}(z) = \begin{cases} 1 & \text{при } z \leq L \\ H(z) & \text{при } -L < z < L \\ -1 & \text{при } z \geq L \end{cases}, \quad H(z) = \begin{cases} z \cdot (\beta - \theta \cdot z) & \text{при } 0 \leq z \leq L \\ z \cdot (\beta + \theta \cdot z) & \text{при } -L \leq z \leq 0 \end{cases}, \quad (4)$$

где z – значение входного сигнала, $\beta = 2/L$, $\theta = 1/L^2$. Интервал L в работе был принят, равной 2.

Погрешность аппроксимации гиперболического тангенса показана на рис. 7.

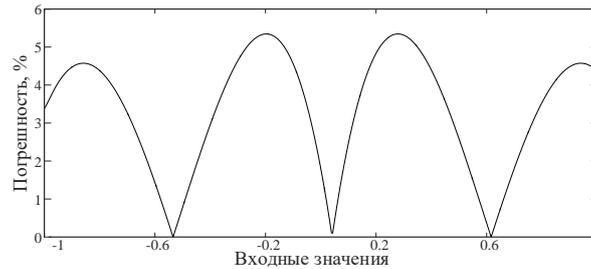


Рис. 7. Погрешность аппроксимации TanH

Максимальная погрешность, связанная с квантованием функции активации TanH, составила -90дБ для 20-битного квантования, -65дБ для 16-битного квантования, -40дБ для 12-битного квантования и -20дБ для 8-битного квантования. Уровень 0 дБ соответствует расхождению значения между идеальным гиперболическим тангенсом и его аппроксимацией на интервале от 0 до 1.

Оценим количество потребляемых ресурсов СНС целевой ПЛИС (Kintex UltraScale KU-040). Доступные ресурсы ПЛИС представлены в табл. 2.

Таблица 2

Доступные ресурсы ПЛИС Kintex UltraScale KU-040

Наименование блоков	Регистры	Число LUTs	Число Block RAM	Число блоков DSPs
Количество	484800	242400	600	1920

Используемые ресурсы ПЛИС Kintex UltraScale KU-040 для СНС с 32-, 20- и 8-битным квантованием представлены на рис. 8.

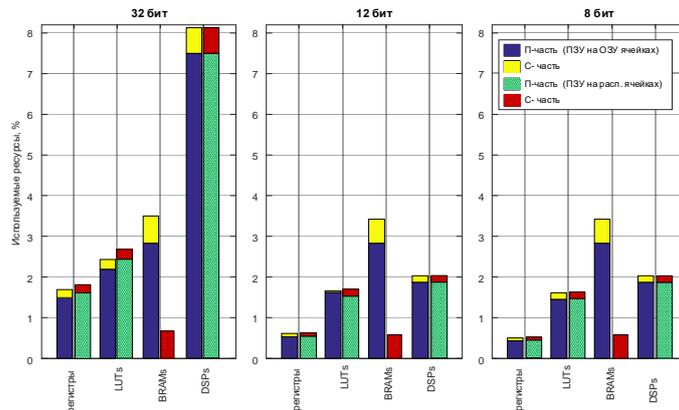


Рис. 8. Используемые ресурсы ПЛИС Kintex UltraScale KU-040

«П-часть» из рисунка означает полносвязную часть нейросети, «С-часть» – сверточную часть нейросети.

Как видно из рис. 8, если реализовать ячейки ПЗУ ПНС на распределенной памяти, то эффективность использования ресурсов ПЛИС увеличится примерно в 4 раза.

В табл. 3 представлена максимальная ошибка квантования относительно вычисления с плавающей запятой (single float) на интервале $[-1;1]$ E_{qMax} . Во втором столбце представлена ошибка для полносвязной части ИНС, в третьем – для всей СНС.

Таблица 3

Погрешности квантования E_{qMax}

Разрядность, бит	ПНС, %	СНС (вместе с ПНС), %
32	4.17e-5	5.96e-5
20	0.0160	0.0228
16	0.27	0.32
12	3.75	4.53
8	27.97	27.97

Причем максимальная ошибка наблюдается в случаях, когда значения на выходах нейросети близки к нулю (т.е., когда на вход подается неклассифицируемое нейросетью изображение).

Из анализа рис. 8 и табл. 3 следует, что понижение уровня квантования до 8 бит нецелесообразно в виду небольшой выгоды в потребляемых ресурсах ПЛИС и значительному повышению погрешности вычислений. Можно сделать вывод, что применение 12-битного квантования СНС является компромиссным решением между ресурсами и погрешностью вычисления.

Заключение. Таким образом была произведена разработка и верификация метода проектирования сверточной нейронной сети для прототипирования на ПЛИС с помощью средств Xilinx System Generator for DSP и Matlab. В рамках данной работы анализировалась простая сверточная нейросеть для входного изображения 6×6 с тремя картами признаков, слоем ПНС с 32-мя нейронами и выходным слоем ПНС с двумя нейронами. Обучение нейросети проводилось с помощью средств Keras.TensorFlow. Коэффициенты фильтров свертки и весовые коэффициенты для полносвязной СНС загружались в Matlab из файла *.h5. Максимальная ошибка квантования на интервале $[-1;1]$ для ИНС с разрядностью 32 бита составила $5e-5$ %, для разрядности 12 бит – 4.53 %. Были проанализированы используемые ресурсы ПЛИС Kintex UltraScale KU-040 разработанной СНС. Реализация СНС с разрядностью 12 бит и ячейками ПЗУ ПНС на распределенной памяти является компромиссным решением между эффективностью использования ресурсов ПЛИС (уменьшение числа блочных ОЗУ BRAM в 4.3 раза и числа блоков ЦОС DSP в 4 раза) и погрешностью квантования (4.5 % относительно вычислений с одинарной точности single float).

Данный метод пригоден для разработки ИНС в проектах по обработке радиолокационного сигнала в радиотехнических системах [19], а также в имитационном моделировании [10]. При реализации СНС по данному методу необходимо учесть, что запись/чтение изображения в ОЗУ наиболее удобно представлять в квадратичной форме. Если же требуется разработать СНС для распознавания линейных сигналов (одномерных), то индексы для записи/чтения сигнала в ОЗУ будут иметь более простой вид, а остальные модули разработанной библиотеки вид не изменят, что говорит об универсальности модулей. Стоит также отметить, что представленный алгоритм двухмерной свертки имеет альтернативы [20], которые можно применять в зависимости от задач, поставленных перед реализацией СНС на ПЛИС (скорость вычисления или ресурсоемкость).

Научные исследования выполнены в рамках проекта «Создание высокотехнологического производства аппаратно-программных комплексов обработки сельхозсырья на основе СВЧ излучения» (Соглашение с Минобрнауки РФ № 075-11-2019-083 от 20.12.2019 г., Договор ЮФУ № 18 от 20.09.2019 г., номер работы в ЮФУ № ХД/19-25-РТ).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Шушцин С.П., Ямаев М.И.* Развитие аппаратно-ориентированных нейронных сетей на FPGA и ASIC // Вестник пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2019. – № 31.
2. *Misra J., Saha I.* Artificial neural networks in hardware: a survey of two decades of progress // Neurocomputing. – 2010. – Vol. 74, No. 1-3, pp. 239-255.
3. *Qiu, J., Song, S., Wang, Y., Yang, H., Wang, J., Yao, S., Xu, N.* Going Deeper with Embedded FPGA Platform for Convolutional Neural Network. Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays – FPGA '16. – 2016. – DOI: 10.1145/2847263.2847265.
4. *Alçın M., Pehlivan İ., Koyuncu İ.* Hardware design and implementation of a novel ANN-based chaotic generator in FPGA // Optik. – 2016. – Vol. 127, No. 13. – P. 5500-5505.
5. *Solovyev R.A. et al.* FPGA implementation of convolutional neural networks with fixed-point calculations // Arxiv preprint arxiv:1808.09945. – 2018.
6. *Воробьев А.Н., Шидловский Д.Ю., Багров А.А.* Аппаратная реализация сверточной нейросети на ПЛИС с использованием модельно-ориентированного проектирования // Цифровая обработка сигналов и её применение – DSPA-2019. – 2019. – С. 423-429.
7. *Bahoura M.* FPGA implementation of an automatic wheezing detection system // Biomedical Signal Processing and Control. – 2018. – Vol. 46. – P. 76-85. (bahoura2018.pdf).
8. *Oniga S.* A new method for FPGA implementation of artificial neural network used in smart devices // International computer science conference microCAD. – 2005. – P. 31-36.
9. *Khodja D.E., Kheldoun A., Refoufi L.* Sigmoid function approximation for ANN implementation in FPGA devices. – 2010.
10. *Бахчевников В.В.* Электродинамическая модель радиосигнала, рассеянного на многослойной структуре, с использованием физической оптики и метода трассировки лучей // Известия высших учебных заведений России. Радиоэлектроника. – 2019. – Т. 22, № 6.
11. *Usenko O.A.* Estimation of diagnostic significance of the control parameters of technological objects based on graph model // Innovative technologies and didactics in teaching. – 2017. – P. 67-73.
12. *Ромм Я.Е., Белоконова С.С., Усенко О.А.* Поиск и идентификация объектов различных типов на основе признаков экстремального вида. II. – 2008.
13. *Design M. B. D. S. P.* Vivado Design Suite Reference Guide. – 2012.
14. *Matworks. Help Center.* Add_block. – URL: https://www.mathworks.com/help/simulink/sref/add_block.html#d120e249911 (accessed 20.04.2020).
15. *Kubat M.* Neural networks: a comprehensive foundation by Simon Haykin, Macmillan, 1994, ISBN 0-02-352781-7 // The Knowledge Engineering Review. – 1999. – Vol. 13, No. 4. – P. 409-412.
16. *Chang J., Sha J.* An efficient implementation of 2D convolution in CNN // IEICE Electronics Express. – 2016. – P. 13.20161134.
17. *Sibi P., Jones S. A., Siddarth P.* Analysis of different activation functions using back propagation neural networks // Journal of Theoretical and Applied Information Technology. – 2013. – Vol. 47, No. 3. – P. 1264-1268.
18. *Panicker M., Babu C.* Efficient FPGA implementation of sigmoid and bipolar sigmoid activation functions for multilayer perceptrons // IOSR J Eng. – 2012. – Vol. 2. – P. 1352-6.
19. *Лобач В.Т. и др.* Последовательный способ формирования каналов ММО при измерении параметров радиолокационных объектов // Известия ЮФУ. Технические науки. – 2015. – № 11 (172). – С. 213-224.
20. *Lu L. et al.* Evaluating fast algorithms for convolutional neural networks on FPGAs // 2017 IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). – IEEE, 2017. – P. 101-108.

REFERENCES

1. Shipitsin S.P., Yamaev M.I. Razvitiye apparatno-orientirovannykh neyronnykh setey na FPGA i ASIC [Development of hardware-oriented neural networks on FPGA and ASIC], *Vestnik permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tekhnologii, sistemy upravleniya* [Bulletin of the Perm national research Polytechnic University. Electrical engineering, information technology, management systems], 2019, No. 31.
2. Misra J., Saha I. Artificial neural networks in hardware: a survey of two decades of progress, *Neurocomputing*, 2010, Vol. 74, No. 1-3, pp. 239-255.
3. Qiu, J., Song, S., Wang, Y., Yang, H., Wang, J., Yao, S., Xu, N. Going Deeper with Embedded FPGA Platform for Convolutional Neural Network, *Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays – FPGA '16*, 2016. DOI: 10.1145/2847263.2847265.
4. Alçın M., Pehlivan İ., Koyuncu İ. Hardware design and implementation of a novel ANN-based chaotic generator in FPGA, *Optik*, 2016, Vol. 127, No. 13, pp. 5500-5505.
5. Solovyev R.A. et al. FPGA implementation of convolutional neural networks with fixed-point calculations, *Arxiv preprint arxiv:1808.09945*, 2018.
6. Vorob'ev A.N., Shidlovskiy D.Yu., Bagrov A.A. Apparattnaya realizatsiya svertochnoy neyroseti na PLIS s ispol'zovaniem model'no-orientirovannogo proektirovaniya [Hardware implementation of convolutional neural network on FPGA using model-oriented design], *Tsifrovaya obrabotka signalov i ee primenenie – DSPA-2019* [Digital signal processing and its application-DSPA-2019], 2019, pp. 423-429.
7. Bahoura M. FPGA implementation of an automatic wheezing detection system, *Biomedical Signal Processing and Control*, 2018, Vol. 46, pp. 76-85. (bahoura2018.pdf).
8. Oniga S. A new method for FPGA implementation of artificial neural network used in smart devices, *International computer science conference microCAD*, 2005, pp. 31-36.
9. Khodja D.E., Kheldoun A., Refoufi L. Sigmoid function approximation for ANN implementation in FPGA devices, 2010.
10. Bakhchevnikov V.V. Elektrodinamicheskaya model' radiosignala, rasseyannogo na mnogosloynnoy strukture, s ispol'zovaniem fizicheskoy optiki i metoda trassirovki luchey [Electrodynamics model of a radio signal scattered on a multilayer structure using physical optics and ray tracing], *Izvestiya vysshikh uchebnykh zavedeniy Rossii. Radioelektronika* [News of higher educational institutions of Russia. Radioelectronics], 2019, Vol. 22, No. 6.
11. Usenko O.A. Estimation of diagnostic significance of the control parameters of technological objects based on graph model, *Innovative technologies and didactics in teaching*, 2017, pp. 67-73.
12. Romm YA.E., Belokonova S.S., Usenko O.A. Poisk i identifikatsiya ob"ektov razlichnykh tipov na osnove priznakov ekstremal'nogo vida. II [Search for and identify various types of objects based on extreme features. II], 2008.
13. Design M. B. D. S. P. Vivado Design Suite Reference Guide, 2012.
14. Matworks. Help Center. Add_block. Available at: https://www.mathworks.com/help/simulink/sref/add_block.html#d120e249911 (accessed 20.04.2020).
15. Kubat M. Neural networks: a comprehensive foundation by Simon Haykin, Macmillan, 1994, ISBN 0-02-352781-7, *The Knowledge Engineering Review*, 1999, Vol. 13, No. 4, pp. 409-412.
16. Chang J., Sha J. An efficient implementation of 2D convolution in CNN, *IEICE Electronics Express*, 2016 33. 13.20161134.
17. Sibi P., Jones S. A., Siddarth P. Analysis of different activation functions using back propagation neural networks, *Journal of Theoretical and Applied Information Technology*, 2013, Vol. 47, No. 3, pp. 1264-1268.
18. Panicker M., Babu C. Efficient FPGA implementation of sigmoid and bipolar sigmoid activation functions for multilayer perceptrons, *IOSR J Eng.*, 2012, Vol. 2, pp. 1352-6.
19. Lobach V.T. i dr. Posledovatel'nyy sposob formirovaniya kanalov MIMO pri izmerenii parametrov radiolokatsionnykh ob"ektov [Sequential method for forming MIMO channels when measuring parameters of radar objects], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2015, No. 11 (172), pp. 213-224.
20. Lu L. et al. Evaluating fast algorithms for convolutional neural networks on FPGAs, 2017 *IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*. IEEE, 2017. pp. 101-108.

Статью рекомендовал к опубликованию д.т.н. А.В. Ковалев.

Бахчевников Валентин Владимирович – Южный федеральный университет; e-mail: bahchevnikov@sfedu.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: +79518289271; кафедра радиотехнических и телекоммуникационных систем; ассистент.

Деркачев Владимир Александрович – e-mail: vderkachev@sfedu.ru; 347922, г. Таганрог, ул. Шевченко, 2; тел.: +79614154733; Научно-конструкторское бюро цифровой обработки сигналов; конструктор.

Бакуменко Алексей Николаевич – e-mail: baku@sfedu.ru; 347900, г. Таганрог, ул. Петровская, 81; тел.: +79886031853; Инжиниринговый центр приборостроения радио и микроэлектроники; инженер.

Bakhchevnikov Valentin Vladimirovich – Southern Federal University; e-mail: bahchevnikov@sfedu.ru; 44, Nekrasovsky, Taganrog, 347928, Russia; phone: +79518289271; the department of radio engineering & telecommunication systems; associate professor.

Derkachev Vladimir Aleksandrovich – e-mail: vderkachev@sfedu.ru; 347922, 2, Shevchenko street, Taganrog, Russia; phone: +79614154733; Research and Design Bureau of Digital Signal Processing; constructor.

Bakumenko Alexey Nikolaevich – e-mail: baku@sfedu.ru; 81, Petrovskaya street, Taganrog, 347900, Russia; phone: +79886031853; Engineering Center of Instrument Making, Radio- and Microelectronics; engineer.

УДК 004.89

DOI 10.18522/2311-3103-2020-3-156-172

**В.В. Бова, Д.Ю. Запорожец, Ю.А. Кравченко, Э.В. Кулиев, В.В. Курейчик,
Н.А. Лызь**

ИДЕНТИФИКАЦИЯ НЕЯВНЫХ УГРОЗ НА ОСНОВЕ АНАЛИЗА АКТИВНОСТИ ПОЛЬЗОВАТЕЛЯ В ИНТЕРНЕТ-ПРОСТРАНСТВЕ*

Статья посвящена проблеме идентификации неявных информационных угроз поисковой деятельности пользователя в Интернет-пространстве на основе анализа его активности в процессе данного взаимодействия. Применение знаний, хранящихся в интернет-пространстве, для реализации преступных намерений несет в себе угрозу для всего общества. Выявление злого умысла в действиях пользователей глобальной информационной сети не всегда является тривиальной задачей. Отработанные технологии анализа контекста интересов пользователя дают сбой в случае осторожных грамотных действий злоумышленников, которые в явном виде не демонстрируют преследуемой ими цели. В работе проведен анализ угроз, связанных с определенными сценариями реализации поисковых процедур, проявляющихся в поисковой деятельности. Описаны критерии оценки неэффективных и эффективного сценариев поиска. Среди признаков, указывающих на возможность наличия угрозы, выделены следующие основные: уход от решения задачи в бесцельную навигацию или к привлекательным ресурсам, поверхностный поиск, отсутствие смыслового погружения в решение поисковой задачи, хаотичные действия при поиске. Для определения наличия неблагоприятных признаков построена система показателей. Сформулированы признаки эффективного сценария организации поиска в Интернет-пространстве, описаны варианты наличия неявных угроз для подобной ситуации. Представлен подход идентификации описанных угроз с учетом заданных критериев оценки различных сценариев поведения пользователя в глобальном информационном пространстве. Разработан алгоритм машинного обучения для идентификации проблемных сценариев путем сравнения с ключевыми паттернами поведения. Создана программная реализация подсистемы идентификации информационных угроз, проведены экспериментальные исследования для подтвержде-

* Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 18-29-22019.