УДК 004.383

С.П. Шипицин, **М.И. Ямаев**

Пермский национальный исследовательский политехнический университет, Пермь, Россия

РАЗВИТИЕ АППАРАТНО-ОРИЕНТИРОВАННЫХ НЕЙРОННЫХ СЕТЕЙ НА FPGA И ASIC

Приводится обзор реализаций нейронных сетей на программируемых логических интегральных схемах (ПЛИС) типа FPGA (Field Programmable Gate Array) и на интегральных схемах специального назначения (Application-Specific Integrated Circuit, ASIC) с 2009 по 2019 г. Приводятся преимущества и ограничения данного подхода при использовании на различных аппаратных платформах. Обозначаются причины целесообразности применения FPGA и ASIC на их основе на той или иной платформе. В частности, FPGA наилучшим образом показывают себя в маломощных мобильных системах, в то время как ASIC, будучи специализированным решением, демонстрируют наибольшую возможную производительность при, однако, слишком высокой цене разработки. Помимо этого проводится сравнение производительности нейронных сетей различных архитектур (перцептроны, свёрточные, бинаризованные, рекуррентные, а также их модификации) на базе FPGA относительно других аппаратных решений по критериям скорости обработки и энергопотребления в соотношении с ценой и простотой развёртывания. Исследуется и по итогам подтверждается высокий интерес к FPGA благодаря высокой энергоэффективности и производительности при решении ряда задач. Показано, что для реализации свёрточных нейронных сетей наилучшим образом подходят графические процессоры, тогда как для рекуррентных — FPGA. Отмечается, что при бинаризации нейронных сетей производительность вентильных матриц значительно повышается, приближаясь к производительности специализированных микросхем. Перспективным направлением последующих исследований является дальнейшее повышение производительности бинаризованных нейронных сетей, реализованных на базе FPGA, путём усовершенствования как математического аппарата, лежащего в основе сети, так и внутренней архитектуры вентильной матрицы и её логических элементов.

Ключевые слова: ПЛИС, ИНС, ППВМ, интегральные схемы специального назначения.

S.P. Shipitsin, **M.I. Iamaev**

Perm National Research Polytechnic University, Perm, Russian Federation

HARDWARE NEURAL NETWORKS PROGRESS ON FPGA AND ASIC

The article provides a survey about the implementation of neural networks on Programmable Logic Device (PLDs) such as FPGA (Field Programmable Gate Array) and Application-Specific Integrated Circuit (ASIC) from 2009 to 2019. Advantages and restrictions of this approach are given using on various hardware platforms. The reasons for the appropriateness to use FPGAs and ASICs based on them on such or another platform are indicated. In particular, FPGAs work best in low-power mobile systems, while ASICs, being a specialized solution, demonstrate the highest possible performance with,

however, the development cost too high. We compare the performance of neural networks of various architectures (perceptrons, convolutional, binarized, recurrent, and also their modifications) based on FPGA with respect to other hardware solutions according to the criteria of processing speed and power consumption in relation to the price and ease of deployment. The high interest in FPGAs is investigated and confirmed as a result of their high energy efficiency and performance in solving a number of tasks. It is shown that GPUs are best suited for the implementation of convolutional neural networks, while FPGAs are suitable for recurrent ones. We note that with binarization of neural networks, the performance of gate arrays increases significantly, approaching the performance of specialized microcircuits. A promising direction for subsequent research is to further increase the performance of binarized neural networks implemented on the basis of FPGA, by improving both the mathematical apparatus underlying the network and the internal architecture of the gate array and its logic elements.

Keywords: programmable logic device, ANN, FPGA, ASIC.

Введение. Производительность и энергоэффективность являются критическими причинами разработки нестандартных архитектур вычислительного оборудования [1]. В задачах искусственного интеллекта, в частности, с использованием аппарата искусственных нейронных сетей (Artificial Neural Networks, ANN), к аппаратному обеспечению предъявляются особенно высокие требования, достаточно сказать, что согласно довольно распространённому среди исследователей мнению прогресс искусственного интеллекта тормозит именно несовершенство аппаратной части. Классические последовательные процессоры не лучшим образом подходят для параллельных вычислений, к которым относятся и нейронные сети. Поскольку FPGA используются для создания конфигурируемых цифровых электронных схем любой архитектуры, в том числе параллельных, то они отлично подходят в качестве ускорителя для создания нейронных сетей. К преимуществам данного подхода можно отнести низкое потребление энергии, высокие вычислительные мощности и гибкость.

1) FPGA могут использоваться для высокоскоростной обработки сигналов, в особенности для датчиков с высокой частотой дискретизации, а также фильтровать данные и снижать скорость передачи данных, что упрощает обработку, передачу и хранение сложного сигнала [2].

2) FPGA в высокой степени подходят для обработки данных в режиме реального времени, таких как изображения, радиолокационные и медицинские сигналы, более эффективно, чем центральные процессоры (Central Processing Unit, CPU) [3].

3) Важными особенностями FPGA являются параллелизм и конвейеризация. Благодаря параллелизму можно многократно распределять и вычислять ресурсы, когда несколько модулей могут работать независимо, одновременно. Конвейеризация делает аппаратные ресур-

сы многоразовыми. Эти два фактора вместе могут значительно улучшить параллельную производительность [4].

Поскольку большинство FPGA реконфигурируемы, это позволяет быстро усовершенствовать и оптимизировать аппаратную реализацию без дополнительных затрат на её производство. Следует отметить, однако, что такая универсализация накладывает ограничения на максимальную производительность, и в ряде задач FPGA не способны на равных конкурировать с процессорами специального назначения. Так, Nurvitadhi и др. (2017) [5] показали, что графические процессоры (Graphics Processing Unit, GPU) выигрывают по скорости у FPGA в операциях над числами с плавающей точкой, хотя при этом проигрывают по эффективности относительно энергопотребления. Авторам удалось достичь сопоставимой скорости обработки чисел с плавающей точкой на Titan X Pascal и Intel Stratix 10. По энергоэффективности FPGA оказалась эффективнее примерно на 40 %, что, впрочем, с учётом стоимости микросхемы преимуществом является очень незначительным.

Misra и Saha (2010) провели обзор исследований в области аппаратно-ориентированных нейронных сетей (HNN) [6]. Авторы отметили возрастающий с конца 1990-х гг. интерес к FPGA. Целью настоящей статьи являются обзор опубликованных в последующие годы исследований аппаратных реализаций нейронных сетей, в частности на FPGA и ASIC, оценка эффективности предложенных подходов и выявление трендов развития отрасли. В табл. 1 показаны примеры аппаратных реализаций различных нейросетевых архитектур, а также подходы к оптимизации таких реализаций.

Таблица 1

Распределение интересов внедрения НС

Тип НС		Аппаратное обеспечение	
		FPGA	ASIC
MLP – перцептрон		[7]	
CNN – свёрточная		[2],[4],[8],[9],[10],[11],[12],[13]	[4],[10]
BNN – бинаризованная		[14],[15],[16]	[14],[15],[17]
RNN	рекуррентная	[18]	
	LSTM	[3],[19],[20]	
	MANN	[21]	
DNN – глубокая НС		[22]	[23]

Свёрточные нейронные сети. Свёрточные нейронные сети (Convolutional Neural Network, CNN) в силу своих архитектурных характеристик демонстрируют самые высокие результаты производительности при реализации на GPU: высокий параллелизм, большое количество однотипных операций, высокие требования к памяти и скорости её работы. GPU с их многочисленными SIMD (Single Instruction, Multiple Data) процессорами и большим объёмом высокоскоростной памяти прекрасно справляются с такими задачами, и улучшить здесь что-то без обращения к самым нижним уровням аппаратной части практически невозможно. Таким образом, CPU проигрывают изначально; FPGA благодаря гибкости позволяют достичь в отдельных случаях более высоких результатов как по скорости работы, так и, конечно, по энергопотреблению, но их относительно высокая цена сводит это преимущество в пользовательских приложениях на нет. Гораздо лучше дела обстоят с ASIC, которые в силу своей изначальной специализации позволяют достигать значительно более высоких результатов; однако они ещё дороже FPGA и часто вообще существуют только на бумаге или в виде имитационных моделей. В течение последнего десятилетия процент публикаций, связанных с реализациями на FPGA и на ASIC, значительно вырос, что связано как со значительным повышением производительности FPGA, так и с применением искусственного интеллекта в IoT-приложениях.

Farabet и др. (2009) [8] предложили реализацию свёрточной сети ConvNet (Convolutional Network) на FPGA, в полной мере реализовав возможность распараллеливания вычислений. Достигнутая скорость распознавания достигла 10 изображений размером 512×384 в секунду на FPGA среднего ценового диапазона. Позднее, в 2011 г., этим же коллективом авторов [4] была предложена свёрточная нейронная сеть гибкой структуры специально для реализации на FPGA. Одним из преимуществ предложенного решения является активное использование «горячей» реконфигурации логической структуры микросхемы. Итоговое сравнение CPU, GPU, ASIC и FPGA показало (за исключением очевидного преимущества ASIC) относительный паритет производительности реализаций на FPGA и на GPU, что подтверждается и более поздними работами, например, Ovtcharov и др. (2015) [9].

T. Chen и др. (2014) [10] разработали нейроморфный ускоритель для крупных CNN и DNN с особым акцентом на оптимизацию использования памяти. Авторам удалось достичь ускорения в 118 раз и снижения энергопотребления в 21 раз по сравнению с SIMD 2ГГц CPU. Для сравнения производительности использовались результаты моделирования, а не реальные микросхемы, что, впрочем, обычно для ASIC. M. Motamedi и др. (2016) [11] предложили архитектуру ускорителя на базе FPGA, который использует все возможные источники параллелизма в глубоких свёрточных нейронных сетях. При этом разработанная модель шаблона архитектуры сети позволяет подобрать оптимальную архитектуру под конкретную аппаратную платформу. Итоговое ускорение в сравнении с прочими реализациями глубоких свёрточных сетей на FPGA достигло 1,9 раза. Подобные инструменты оптимизации предлагали и другие коллективы исследователей. Так, Wang и др. (2016) [12] разработали инструмент автоматической реализации Caffe CNN на FPGA специально для устройств с малым энергопотреблением. DiCeccо и др. в 2016 г. [13] предприняли попытку частичного переноса расчётов CNN (одного свёрточного слоя) с CPU на FPGA, правда, неудачную.

В качестве конкретного практического примера можно упомянуть работу Zhao и др. (2019) [2]. Авторы реализовали распознавание изображений для автономного подводного робота на базе FPGA Xilinx Zync 7035 в связи с необходимостью обеспечить приемлемую точность при минимальном энергопотреблении. В результате применения ряда оптимизаций CNN под FPGA ценой небольшой потери точности распознавания удалось добиться обработки изображений разрешения 1920×1080 в реальном времени (с частотой 25 fps) при энергопотреблении 9,5 Вт.

Рекуррентные нейронные сети. Рекуррентные нейронные сети (Recurrent Neural Network, RNN) имеют связи, образующие направленную последовательность, а потому могут лучше работать на CPU, нежели GPU, которые теряют здесь своё преимущество параллелизма. В архитектуре потока данных последние передаются напрямую от одного элемента обработки к другому, что снижает потребность в энергопотребляющих доступах к памяти. Именно поэтому FPGA хорошо подходят для внедрения RNN [21].

Li Sicheng и др. (2015) [18] предложили набор улучшений для языковой модели на основе рекуррентной сети (RNNLM) при реализации на FPGA: расширение параллелизма, введение вычислений разной точности, оптимизация работы с памятью, масштабируемость. Задействовав менее половины ресурсов кристалла Virtex-6, им удалось приблизиться к качеству работы лучших на тот момент тестов, где в отличие от их работы использовался препроцессинг. И хотя не удалось превзойти по скорости NVIDIA GeForce GTX 580 (GPU), прирост по энергоэффективности оказался в 6,68 раза.

Park и др. (2015) [21] реализовали так называемую MANN (дополненная памятью нейронная сеть, Memory-Augmented Neural Network) на FPGA, добавив к этому свой метод порогового отсечения вывода. Ускоритель получился в среднем в 125 раз более энергоэффективным, чем GPU NVIDIA TITAN V. Более того, реализация предложенного метода увеличила преимущество в среднем до 140 раз, хотя показанные результаты нестабильны. FPGA оказалась эффективнее в 5,2–8,0 в разных тестах и на разном железе. Наибольший прирост предложенный метод дал на низких частотах. Во всех случаях использовался один и тот же ускоритель без оптимизаций к конкретному железу.

Chang и др. (2015) [3] реализовали LSTM (сети долгой краткосрочной памяти, Long Short Term Memory) на FPGA Xilinx Zynq 7020. Реализация оказалась в 21 раз быстрее, чем ARM Cortex-A9 (CPU), который внедрён в Zynq 7020, что свидетельствует о том, что FPGA не только эффективнее, но и быстрее CPU в случае LSTM. В свою очередь, Lee и др. (2016) [19] предложили систему обработки естественного языка на основе двух LSTM-RNN для обработки различных этапов распознавания речи. Для улучшения производительности применена также статистическая модель слов. Результаты работы объединяются и сравниваются посредством алгоритма N-лучших. Реализована эта архитектура на FPGA, что позволило достигнуть скорости, превышающей реальное время в 4,12 раза. На однопоточном режиме это превышает результаты GPU NVIDIA GeForce Titan X (3,36). Энергопотребление составило 9,24 Вт против около 80 Вт у GPU, а итоговое качество распознавания несколько ниже, чем в других работах, однако это компенсируется компактностью и скоростью предложенной модели. Вообще, среди исследователей отмечается довольно большое внимание к LSTM, так, Guan Yijin и др. (2017) [24] предлагают ускоритель

LSTM-RNN на базе FPGA. Их работа так же демонстрирует значительное ускорение в сравнении с CPU и высокую энергоэффективность.

Guo Kaiyuan и др. (2016) [20] предложили рабочий процесс быстрой реализации нейросетевых архитектур на FPGA. В частности, Aristotle и Descartes – архитектуры для реализации на FPGA соответственно CNN и разрежённых LSTM. Предложенный инструмент автоматической компиляции позволяет не реализовывать архитектуры нейросетей на OpenCL вручную, а просто скомпилировать код, написанный для высокоуровневых фреймворков типа TensorFlow. Оценки производительности впечатляют, поэтому архитектуры рекомендуются к использованию.

Бинаризованные нейронные сети. Бинаризованные нейронные сети (Binary Neural Networks, BNN) – это перспективный и очень многообещающий подход для оптимизации архитектур НС для их реализации на FPGA. Он был предложен в 2015 г. [25], но уже получил заслуженное внимание со стороны исследователей благодаря значительным достигаемым преимуществам. Суть данного подхода состоит в том, чтобы заменить часть параметров сети (входные веса нейронов, внутренние значения нейронов), которые изначально представлены числами с плавающей или фиксированной точкой, бинарными значениями. Ускоритель получает значительный прирост по скорости и энергоэффективности ценой незначительной потери точности. Такие сети, очевидно, подходят не для всех задач, однако значительно улучшают показатели энергоэффективности аппаратного обеспечения, делая FPGA уже вполне подходящим решением для мобильных приложений.

Andri и др. (2016) [17], признавая повышение производительности CNN в последние годы, отметили их высокое энергопотребление и недоступность для мобильных устройств даже в случае использования ASIC. Поэтому они предложили первую оптимизированную разработку, в которой реализован гибкий, энергоэффективный и масштабируемый свёрточный механизм, основанный на Binary Connect CNN. Разработанный ускоритель достиг эффективности в 1510 GOp/s, что быстрее в 2,7 раза и в 35 раз энергоэффективнее, чем лучший на тот момент ASIC.

Nurvitadhi и др. (2016) [14] исследовали разные типы ускорителей для BNN. И даже если FPGA могут опережать по энергоэффективности CPU и GPU, то при этом 14nm ASIC сейчас более эффективен, чем Stratix V и Arria 10. Им удалось добиться 9,8 TOP/sec на Arria 10

с архитектурой из 1024 блоков обработки, что оказалось довольно близко к ASIC. Разрыв в эффективности относительно ASIC может стать меньше с более новыми FPGA, с более жёсткими блоками ЦОС, встроенными DRAM и более высокой частотой.

Zhao и др. (2017) [15] протестировали и сравнили реализации бинаризованных CNN на FPGA. Нейронные сети обрабатывались и обучались на наборе данных CIFAR-10. При сравнении разных ускорителей Xilinx Zynq-7000 SoC (FPGA) оказалась почти в два раза быстрее Intel Xeon E5-2640 (CPU), но медленнее примерно в 8 раз, чем NVIDIA Tesla K40 (GPU). Все ускорители сравнивались относительно встроенной платы NVIDIA Jetson TK1 (mGPU). При этом по относительной характеристике с учётом энергопотребления FPGA оказалась в 7 раз эффективнее, чем GPU.

Liang и др. (2018) [16] предложили новую технологию оптимизации нейронных сетей под FPGA – FP-BNN. На основе разработанного комплекса мер авторы бинаризировали MNIST MLP, Cifar-10 ConvNet, AlexNet в их реализации на FPGA и демонстрируют преимущества предложенного подхода. Сети получилось сжать в среднем в 32 раза относительно первоначального размера. Эффективность в сравнении с CPU и GPU составила от 2,72 раза на AlexNet до 314 раз на MNIST MLP.

Прочие подходы. Благодаря большим практическим успехам рекуррентные и свёрточные сети получили самое большое распространение и развитие, но по другим нейросетевым архитектурам также публикуются отдельные работы. Gomperts и др. (2011) [7] предложили реализацию многослойного перцептрона на FPGA, обладающего способностью дообучения с помощью метода обратного распространения ошибки во время работы. При итоговом тестировании качество работы оказалось не очень стабильным и иногда даже ухудшалось после дообучения. Впрочем, в тех приложениях, где дообучение на ходу необходимо, а реализация генеративно-состязательной сети (GAN) нецелесообразна, такой подход может оказаться полезным.

Кроме специализированных оптимизаций различных нейросетевых архитектур на конкретном аппаратном обеспечении исследователи предлагают общие подходы, позволяющие минимизировать энергопотребление микросхемы при сохранении точности работы сети. Так, B. Reagen и др. (2016) [23] предложили подход к совместной разработке ускорителей глубоких нейронных сетей на разных уровнях: алго-

ритмическом, архитектурном и схемотехническом, чтобы оптимизировать характеристики получаемой сети и ASIC под эту сеть, добиваясь значительного (до 8,1 раза по сравнению с базовой моделью) снижения энергопотребления. Ранее они же в 2014 г. [26] разработали и представили Aladdin – пре-RTL симулятор специализированных ускорителей нейронных сетей, позволяющий, в отличие от существующих решений, благодаря незначительному снижению точности симуляции повысить её скорость более чем в 100 раз, обеспечивая возможность моделирования чипов куда большего размера. Zhang Xiaofan и др. (2017) [22] предложили фреймворк для анализа системных требований глубоких нейронных сетей, оптимизированных под реализацию на FPGA. Цель – максимальное ускорение работы сети путём оптимизации работы с памятью, уменьшения количества внутренних циклов, квантования и сокращения самой сети. Для тестирования они использовали вариацию ConvLSTM (совмещение CNN и RNN). При сравнении реализаций сети на FPGA (Xilinx XC7VX690T) с GPU (NVIDIA K80) и CPU (Intel Xeon E5-2630) были достигнуты прирост скорости в 4,75 раза и снижение энергопотребления в 17,5 раз.

Обсуждение и выводы. Исследования последних лет (табл. 2) показывают, что FPGA являются значительно более энергоэффективными, чем прочие аппаратные решения. Причем относительное снижение энергопотребления может составлять до двух порядков.

Таблица 2

Сравнение энергоэффективности аппаратных ускорителей ИНС

Статья	Park и др. (2017) [21]	Han и др. (2016) [27]	Zhang и др. (2017) [22]	Zhao и др. (2017) [15]	Wang и др. (2016) [12]
Метрика	FLOPS/kJ	GOP/W	imgs./J	imgs/sec/W	J^{-1}
CPU	1,70x	1,0x	1,00x	1,0x	1,0x
GPU	1,00x	13,8x/3,5x	1,01x	8,2x	–
FPGA	139,75x	197,0x/40,0x	17,89x	50,4x	58,0x
mGPU	–	–	–	1,4x	–

На данный момент, как представляется, FPGA целесообразно применять либо в небольших (или даже встроенных) системах, либо в высоконагруженных облачных сервисах. В первом случае FPGA маломощны, но и относительно недороги – при неплохой производительности в носимой электронике и встроенных системах это лучший

выбор. Самые же производительные FPGA стоят существенно дороже GPU сопоставимой производительности, но при этом достаточно надёжны и куда более экономичны с энергетической точки зрения. В случае использования FPGA для мобильных устройств конкурентным решением могут считаться адаптированные под такие задачи мобильные процессоры (mGPU), например, семейство ускорителей NVIDIA Jetson. Однако в статье [15] показано, что фактическая производительность не соответствует заявленным характеристикам из-за проблем с многопоточной обработкой данных. Более того, возможность реконфигурации FPGA позволяет использовать их в самых разных задачах в зависимости от текущих потребностей, что позволяет использовать ресурсы микросхемы с максимальной эффективностью.

Достаточно высокая цена делает FPGA не лучшим выбором для реализации свёрточных сетей, даже с учётом сопоставимой с GPU производительности. С другой стороны, в случае с рекуррентными сетями ситуация кардинально отличается: тут они могут превосходить GPU на два порядка. При этом предложенный в 2015 г. метод бинаризации нейронных сетей приближает FPGA к ASIC как по производительности, так и по энергопотреблению при незначительной потере точности. Таким образом, FPGA становятся незаменимым аппаратным решением при внедрении приложений искусственного интеллекта на мобильные устройства и в IoT.

Бинаризованные нейронные сети являются наилучшим кандидатом для реализации на FPGA благодаря заложенной в их архитектуре двоичности, несмотря на потери точности в сравнении с прочими архитектурами. В отдельных случаях оптимизации делают ускоритель на базе FPGA сопоставимым по производительности с принципиальным максимумом -– ASIC. Тем не менее, эта планка до сих пор не достигнута (FPGA показывают худшие в 5–10 раз результаты), и возможны значительные усовершенствования. При этом, что немаловажно, гибкость вентильных матриц позволяет как достаточно быстро разрабатывать решения на их базе, так и менять внутреннюю конфигурацию чипа под изменившиеся требования программными средствами, что даёт им существенное преимущество универсальности в сравнении с аппаратно жёсткими специализированными микросхемами. Таким образом, оптимальным направлением дальнейших исследований представляется повышение производительности бинаризованных нейронных сетей,

реализованных на базе FPGA, путём усовершенствования как математического аппарата, лежащего в основе сети, так и внутренней архитектуры вентильной матрицы и её логических элементов.

Библиографический список

1. Тюрин С.Ф., Плотникова А.Ю. Концепция «зеленой» логики // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2013. – № 8.

2. Real-Time Underwater Image Recognition with FPGA Embedded System for Convolutional Neural Network / M. Zhao [et al.] // Sensors. – 2019. – Vol. 19. – № 2. – P. 350.

3. Chang A.X.M., Martini B., Culurciello E. Recurrent neural networks hardware implementation on FPGA: arXiv preprint arXiv:1511.05552. – 2015.

4. Large-scale FPGA-based convolutional networks / C. Farabet [et al.] // Scaling up Machine Learning: Parallel and Distributed Approaches. – 2011. – P. 399–419.

5. Can FPGAs beat GPUs in accelerating next-generation deep neural networks? / E. Nurvitadhi [et al.] // Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. – ACM, 2017. – P. 5–14.

6. Misra J., Saha I. Artificial neural networks in hardware: A survey of two decades of progress // Neurocomputing. – 2010. – Vol. 74. – № 1–3. – P. 239–255.

7. Gomperts A., Ukil A., Zurfluh F. Development and implementation of parameterized FPGA-based general purpose neural networks for online applications // IEEE Transactions on Industrial Informatics. – 2011. – Vol. 7. – № 1. – P. 78–89.

8. Cnp: An FPGA-based processor for convolutional networks / C. Farabet [et al.] // Field Programmable Logic and Applications: International Conference. 2009. FPL 2009. – IEEE, 2009. – P. 32–37.

9. Accelerating deep convolutional neural networks using specialized hardware / K. Ovtcharov [et al.] // Microsoft Research Whitepaper. – 2015. – Vol. 2. – № 11.

10. Diannao: A small-footprint high-throughput accelerator for ubiquitous machine-learning / T. Chen [et al.] // ACM Sigplan Notices. – ACM, 2014. – Vol. 49. – Diannao. – P. 269–284.

11. Design space exploration of fpga-based deep convolutional neural networks / M. Motamedi [et al.] // 21st Asia and South Pacific Design Automation Conference (ASP-DAC). – IEEE, 2016. – P. 575–580.

12. DeepBurning: automatic generation of FPGA-based learning accelerators for the neural network family / Y. Wang [et al.] // Proceedings of the 53rd Annual Design Automation Conference. – ACM, 2016. – DeepBurning. – P. 110.

13. Caffeinated FPGAs: FPGA framework for convolutional neural networks / R. DiCecco [et al.] // International Conference on Field-Programmable Technology (FPT). – IEEE, 2016. – Caffeinated FPGAs. – P. 265–268.

14. Accelerating binarized neural networks: Comparison of FPGA, CPU, GPU, and ASIC / E. Nurvitadhi [et al.] // International Conference on Field-Programmable Technology (FPT). – IEEE, 2016. – Accelerating binarized neural networks. – P. 77–84.

15. Accelerating binarized convolutional neural networks with software-programmable fpgas / R. Zhao [et al.] // Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. – ACM, 2017. – P. 15–24.

16. FP-BNN: Binarized neural network on FPGA / S. Liang [et al.] // Neurocomputing. – 2018. – Vol. 275. – FP-BNN. – P. 1072–1086.

17. YodaNN: An ultra-low power convolutional neural network accelerator based on binary weights / R. Andri [et al.] // IEEE Computer Society Annual Symposium on VLSI (ISVLSI). – IEEE, 2016. – YodaNN. – P. 236–241.

18. Fpga acceleration of recurrent neural network based language model / S. Li [et al.] // IEEE 23rd Annual International Symposium on Field-Programmable Custom Computing Machines. – IEEE, 2015. – P. 111–118.

19. FPGA-based low-power speech recognition with recurrent neural networks / M. Lee [et al.] // IEEE International Workshop on Signal Processing Systems (SiPS). – IEEE, 2016. – P. 230–235.

20. From model to FPGA: Software-hardware co-design for efficient neural network acceleration / K. Guo [et al.] // IEEE Hot Chips 28 Symposium (HCS). – IEEE, 2016. – From model to FPGA. – P. 1–27.

21. Energy-efficient inference accelerator for memory-augmented neural networks on an FPGA / S. Park [et al.] // Design, Automation & Test in Europe Conference & Exhibition (DATE). – IEEE, 2019. – P. 1587–1590.

22. High-performance video content recognition with long-term recurrent convolutional network for FPGA / X. Zhang [et al.] // 27th International Conference on Field Programmable Logic and Applications (FPL). – IEEE, 2017. – P. 1–4.

23. Minerva: Enabling low-power, highly-accurate deep neural network accelerators / B. Reagen [et al.] // ACM SIGARCH Computer Architecture News. – IEEE Press, 2016. – Vol. 44. – Minerva. – P. 267–278.

24. FPGA-based accelerator for long short-term memory recurrent neural networks / Y. Guan [et al.] // 22nd Asia and South Pacific Design Automation Conference (ASP-DAC). – IEEE, 2017. – P. 629–634.

25. Courbariaux M., Bengio Y., David J.-P. Binaryconnect: Training deep neural networks with binary weights during propagations // Advances in neural information processing systems. – 2015. – Binaryconnect. – P. 3123–3131.

26. Aladdin: A pre-rtl, power-performance accelerator simulator enabling large design space exploration of customized architectures / Y.S. Shao [et al.] // ACM SIGARCH Computer Architecture News. – IEEE Press, 2014. – Vol. 42. – Aladdin. – P. 97–108.

27. Ese: Efficient speech recognition engine with sparse lstm on fpga / S. Han [et al.] // Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. – ACM, 2017. – Ese. – P. 75–84.

References

1. Tiurin S.F., Plotnikova A.Iu. Kontseptsiia «zelenoi» logiki [The concept of green logic]. *Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotekhnika, informatsionnye tekhnologii, sistemy upravleniia*, 2013, no. 8.

2. Zhao M. et al. Real-Time Underwater Image Recognition with FPGA Embedded System for Convolutional Neural Network. *Sensors*, 2019, vol. 19, no. 2, P. 350.

3. Chang A.X.M., Martini B., Culurciello E. Recurrent neural networks hardware implementation on FPGA: arXiv preprint arXiv:1511.05552, 2015.

4. Farabet C. et al. Large-scale FPGA-based convolutional networks. *Scaling up Machine Learning: Parallel and Distributed Approaches*, 2011, pp. 399-419.

5. Nurvitadhi E. et al. Can FPGAs beat GPUs in accelerating next-generation deep neural networks? *Proceedings of the 2017 ACM/SIGDA*

International Symposium on Field-Programmable Gate Arrays. ACM, 2017, pp. 5-14.

6. 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.

7. Gomperts A., Ukil A., Zurfluh F. Development and implementation of parameterized FPGA-based general purpose neural networks for online applications. IEEE *Transactions on Industrial Informatics*, 2011, vol. 7, no. 1, pp. 78-89.

8. Farabet C. et al. Cnp: An FPGA-based processor for convolutional networks. *Field Programmable Logic and Applications: International Conference. 2009. FPL 2009.* IEEE, 2009, pp. 32-37.

9. Ovtcharov K. et al. Accelerating deep convolutional neural networks using specialized hardware. *Microsoft Research Whitepaper*, 2015, vol. 2, no. 11.

10. Chen T. et al. Diannao: A small-footprint high-throughput accelerator for ubiquitous machine-learning. *ACM Sigplan Notices*. ACM, 2014, vol. 49, Diannao, pp. 269-284.

11. Motamedi M. et al. Design space exploration of fpga-based deep convolutional neural networks. *21st Asia and South Pacific Design Automation Conference (ASP-DAC).* IEEE, 2016, pp. 575-580.

12. Wang Y. et al. DeepBurning: automatic generation of FPGA-based learning accelerators for the neural network family. *Proceedings of the 53rd Annual Design Automation Conference.* ACM, 2016. DeepBurning. P. 110.

13. DiCecco R. et al. Caffeinated FPGAs: FPGA framework for convolutional neural networks. I*nternational Conference on Field-Programmable Technology (FPT).* IEEE, 2016. Caffeinated FPGAs, pp. 265-268.

14. Nurvitadhi E. et al. Accelerating binarized neural networks: Comparison of FPGA, CPU, GPU, and ASIC. *International Conference on Field-Programmable Technology (FPT).* IEEE, 2016. Accelerating binarized neural networks, pp. 77-84.

15. Zhao R. et al. Accelerating binarized convolutional neural networks with software-programmable fpgas. *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays.* ACM, 2017, pp. 15-24.

16. Liang S. et al. FP-BNN: Binarized neural network on FPGA. *Neurocomputing*, 2018, vol. 275, FP-BNN, pp. 1072-1086.

17. Andri R. et al. YodaNN: An ultra-low power convolutional neural network accelerator based on binary weights. *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 2016, YodaNN, pp. 236-241.

18. Li S. et al. Fpga acceleration of recurrent neural network based language model. *IEEE 23rd Annual International Symposium on Field-Programmable Custom Computing Machines*. IEEE, 2015, pp. 111-118.

19. Lee M. et al. FPGA-based low-power speech recognition with recurrent neural networks. *IEEE International Workshop on Signal Processing Systems (SiPS)*. IEEE, 2016, pp. 230-235.

20. Guo K. et al. From model to FPGA: Software-hardware co-design for efficient neural network acceleration. *IEEE Hot Chips 28 Symposium (HCS)*. IEEE, 2016, From model to FPGA, pp. 1-27.

21. Park S. et al. Energy-efficient inference accelerator for memory-augmented neural networks on an FPGA. *Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2019, pp. 1587-1590.

22. Zhang X. et al. High-performance video content recognition with long-term recurrent convolutional network for FPGA. *27th International Conference on Field Programmable Logic and Applications (FPL)*. IEEE, 2017, pp. 1-4.

23. Reagen B. et al. Minerva: Enabling low-power, highly-accurate deep neural network accelerators. *ACM SIGARCH Computer Architecture News*. IEEE Press, 2016, vol. 44, Minerva, pp. 267-278.

24. Guan Y. et al. FPGA-based accelerator for long short-term memory recurrent neural networks. *22nd Asia and South Pacific Design Automation Conference (ASP-DAC)*. IEEE, 2017, pp. 629-634.

25. Courbariaux M., Bengio Y., David J.-P. Binaryconnect: Training deep neural networks with binary weights during propagations. *Advances in neural information processing systems*, 2015, Binaryconnect, pp. 3123-3131.

26. Shao Y.S. et al. Aladdin: A pre-rtl, power-performance accelerator simulator enabling large design space exploration of customized architectures. *ACM SIGARCH Computer Architecture News*. IEEE Press, 2014, vol. 42, Aladdin, pp. 97-108.

27. *Han S. et al. Ese: Efficient speech recognition engine with sparse lstm on fpga. Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*. ACM, 2017, Ese, pp. 75-84.

Сведения об авторах

Шипицин Сергей Павлович (Пермь, Россия) – аспирант кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета (614990, Пермь, Комсомольский пр., 29, e-mail: s.p.shipitsin@gmail.com).

Ямаев Марсель Ильгамович (Пермь, Россия) – аспирант кафедры «Автоматика и Телемеханика» Пермского национального исследовательского политехнического университета (614990, Пермь, Комсомольский пр., 29, e-mail: marsel.iamaev@mail.ru).

About the authors

Shipitsin Sergei Pavlovich (Perm, Russian Federation) is a Graduate Student Department of Automation and Telemechanics Perm National Research Polytechnic University (614990, Perm, 29, Komsomolsky pr., e-mail: s.p.shipitsin@gmail.com).

Iamaev Marsel Ilgamovich (Perm, Russian Federation) is a Graduate Student Department of Automation and Telemechanics Perm National Research Polytechnic University (614990, Perm, 29, Komsomolsky pr., e-mail: marsel.iamaev@mail.ru).

Получено: 17.07.2019