Раздел II. Методы, модели и алгоритмы обработки информации

УДК 004.8 DOI 10.18522/2311-3103-2021-6-64-72

В.В. Ковалев, Н.Е. Сергеев

РЕАЛИЗАЦИЯ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ НА ВСТРАИВАЕМЫХ УСТРОЙСТВАХ С ОГРАНИЧЕННЫМ ВЫЧИСЛИТЕЛЬНЫМ РЕСУРСОМ

Большие объемы видеоданных, фиксируемые сенсорными датчиками в различных спектральных диапазонах, существенные размеры архитектур сверточных нейронных сетей создают проблемы с реализацией нейросетевых алгоритмов на периферийных устройствах из-за значительных ограничений вычислительных ресурсов на встраиваемых вычислительных устройствах. В статье рассмотрено применение алгоритмов автоматического поиска и распознавания образов на основе методов машинного обучения, реализованных на встраиваемых устройствах с вычислительным ресурсом Graphics Processing Unit. В качестве алгоритма поиска и распознавания образов используются детекционные сверточные нейронные сети «You Only Look Once V3» и «You Only Look Once V3-Tiny», которые реализованы на встраиваемых вычислительных устройствах линейки NVIDIA Jetson, находящиеся в разном ценовом диапазоне и с различным вычислительным ресурсом. Также в работе экспериментальным путем вычислены оценки алгоритмов на встраиваемых устройствах по таким показателям, как потребляемая мощность, время прямого прохода сверточнной нейронной сети и точность обнаружения. На основе решений реализованных, как на аппаратном уровне, так и на программном, представляющихся компанией NVIDIA становится возможным применение глубоких нейросетевых алгоритмов на основе операции свертка в режиме реального времени. Рассмотрены методы оптимизации вычислений, предлагаемые компанией NVIDIA. Произведены экспериментальное исследования влияния вычислений с пониженной точностью на скорость работы и точность обнаружения объектов на изображениях, исследуемых архитектур сверточных нейронных сетей, которые были предварительно обучены на выборке изображений состоящей из датасетов PASCAL VOC 2007 и PASCAL VOC 2012.

Сверточные нейронный сети; оптимизация вычислений; встраиваемые вычислительные устройства; методы оптимизации; обнаружения объектов.

V.V. Kovalev, N.E. Sergeev

IMPLEMENTATION OF CONVENTIONAL NEURAL NETWORKS ON EMBEDDED DEVICES WITH A LIMITED COMPUTING RESOURCE

Large amounts of video data captured by sensor sensors in various spectral ranges, the significant size of convolutional neural network architectures create problems with the implementation of neural network algorithms on peripheral devices due to significant limitations of computing resources on embedded computing devices. The article discusses the use of algorithms for automatic search and pattern recognition based on machine learning methods, implemented on embedded devices with a computing resource Graphics Processing Unit. Detection convolutional neural networks «You Only Look Once V3» and «You Only Look Once V3-Tiny» are used as a search and pattern recognition algorithm, which are implemented on embedded computing devices of the NVIDIA Jetson line, located in different price ranges and with different computing resources ... Also, in the work, the estimates of

algorithms on embedded devices are experimentally calculated for such indicators as power consumption, forward passage time of a convolutional neural network, and detection accuracy. On the basis of solutions implemented, both at the hardware level and in software, presented by NVIDIA, it becomes possible to use deep neural network algorithms based on the convolution operation in real time. Computational optimization methods offered by NVIDIA are considered. Experimental studies of the influence of computations with reduced accuracy on the speed and accuracy of object detection in images of the investigated architectures of convolutional neural networks, which were previously trained on a sample of images consisting of the PASCAL VOC 2007 and PASCAL VOC 2012 datasets, have been carried out.

Convolutional neural networks; computing optimization; embedded computing devices; optimization methods; object detection.

Введение. Автоматизация процессов в таких отраслях, как медицина, авиация, производство, логистика и т.д. на сегодняшний момент, является одним из важных и активно развивающимся в мире технологическим направлением. Для обеспечения корректной работы автоматизированных систем сенсорные датчики считывают большие объемы данных в различных спектральных диапазонах, создающих высокоскоростные потоки данных, требующих обработки в режиме реального времени. Одной из востребованных задач обработки данных являются автоматический поиск, классификация и распознавания объектов в потоке видеокадров. Данные алгоритмы применяются и для разработки систем автоматического управления, распознавания, ситуационной осведомленности, автономных робототехнических и других систем. С практической точки зрения для решения подобных классов задач хорошо себя зарекомендовали методы и алгоритмы на основе искусственного интеллекта [1], которые отличаются возможностями адаптации и улучшения функционирования алгоритмов в результате обучения/дообучения систем на новых расширенных представительных наборах данных. В настоящее время такие алгоритмы активно развиваются на основе методов машинного обучения глубоких нейронных сетей, среди которых широко применяются архитектуры сверточных нейронных сетей.

Основной объем требований к производительности вычислительных ресурсов встраиваемых устройств предъявляют к обработке изображений высокого разрешения. Такие изображения могут формироваться сенсорами различных спектральных диапазонов, включая видимый, инфракрасный, радиолокационный и другие диапазоны, и содержать порядка миллиона элементов изображения (пикселей). Обработка таких изображений в масштабе реального времени, как правило, должна выполняться за время порядка $10-100$ [мс], что определяет необходимость достижения скорости обработки пикселей порядка 10^7-10^8 [пикселей/с]. Встроенные современные вычислительные устройства с малым форм-фактором позволяют выполнять большое количество параллельных вычислений благодаря наличию мощных сопроцессоров, что позволяет применять устройства для решения задач обработки видеоданных. Роль сопроцессоров в них выполняют GPU или FPGA с малым форм фактором.

Сложность и размер нейронных сетей продолжают расти. Новые сети с тысячами слоев и миллионами нейронов требуют еще более высокой производительности и более быстрого обучения. Это накладывает определенные требования на используемые ресурсы, такие как время отклика, размер используемой памяти и потребляемая мощность. Поэтому задача оптимизации нейросетевых алгоритмов является актуальной в настоящее время.

1. Обзор встраиваемых вычислительных устройств для реализации методов искусственного интеллекта. ***1.1. Центральный процессор Central Processing Unit (CPU).*** CPU – центральное обрабатывающее устройство персонально-

го компьютера. Одним из лидеров в производительности процессоров является компания Intel, которая поставляет наряду с процессорами библиотеку для ускорения нейронных сетей oneAPI DL Framework Developer Toolkit.

CPU обладает высокой программируемостью, поддерживает большое количество типов данных, поддерживает SIMD инструкции для параллельных вычислений, позволяет эффективно использовать свойства разреженности весов нейронной сети.

CPU имеет низкую производительность, потому что вычисления не направлены на выполнения однотипных операций. Например, модель CPU Intel Xeon E7-8860 v4 имеет пиковую производительность, равную 283 GFLOPs.

Реализация СНС на процессорах CPU не позволяет реализовать большое количество параллельных вычислений однотипных операций, свойственным нейросетевым алгоритмам из-за небольшого количества вычислительных ядер.

Большинство процессоров оптимизированы под вычисление целочисленных операций. Поэтому пути оптимизации основаны на:

- минимизации вычислительной сложности архитектуры СНС;
- квантование и перевод весовых коэффициентов СНС в целочисленные значения (INT8).

Исходя из этого, целесообразно применять архитектуры СНС основанные на базисе поканальной свертки (depthwise convolution), что существенно снижает вычислительную сложность алгоритма в целом.

1.2. Программируемая логическая интегральная схема Field Programmable Gate Array. Программируемая логическая интегральная схема – устройство, на котором реализация нейронной сети осуществляется на аппаратном уровне. FPGA поддерживают типы данных с произвольным количеством бит, потенциально возможна высокая производительность, имеют низкую потребляемую мощность. К недостаткам следует отнести низкую скорость программирования, высокая трудоемкость внесения изменений в конфигурацию ПЛИС. В большинстве случаев процесс обучения нейронной сети на ПЛИС не представляется возможным из-за малого объема памяти.

1.3. Tensor Processing Unit. TPU (Tensor Processing Unit) – компания Google представила в 2017 году специализированную интегральную схему (ASICs) для ускорения нейросетевых вычислений. Матричный процессор способен выполнять сотни тысяч операций типа умножение за один такт, содержит блок активации, унифицированный буфер, оптимизирован с точки зрения обращения к DRAM, поддерживает квантование в 8 бит в целой точке, набор инструкций CISC. Если обучение нейронных сетей с большим количеством слоев может занимать недели на GPU, то на TPU это займет всего лишь часы.

Достоинства: TPU показывает очень высокую производительность, особенно для больших значений размера батча (batch size) и нейронных сетей большого размера, имеет меньшую потребляемую мощность по сравнению с GPU. Пиковая производительность TPU может достигать 92 TOPs.

Недостатки: проект должен быть реализован только с помощью библиотеки TensorFlow, для прямого прохода нейронной сети с одним изображением показывает меньшую производительность по сравнению с GPU, обладают высокой стоимостью.

1.4. Графический процессор Graphics Processing Unit (GPU). Широкое применение в решении задач обработки видеоданных осуществляется с применением встраиваемых вычислительных устройств линейки NVIDIA Jetson на базе искусственного интеллекта (ИИ) [2].

Модуль NVIDIA Jetson Nano – это компактный компьютер на базе ИИ, который обладает производительностью и энергоэффективностью, необходимой для современных задач искусственного интеллекта, параллельной работы нескольких нейронных сетей и одновременной обработки данных с нескольких датчиков высокого разрешения. Поэтому он представляет собой отличное начальное решение для внедрения сложных средств искусственного интеллекта во встраиваемые системы [3].

Встраиваемые модули NVIDIA Jetson TX2 обеспечивают производительность до 2,5 раз больше, чем Jetson Nano, при уровне энергопотребления в 7,5 Вт. Jetson TX2 NX совместим с Jetson Nano по выводам и форм-фактору, а Jetson TX2, TX2 4GB и TX2i имеют оригинальный форм-фактор Jetson TX2. Надежный модуль Jetson TX2i идеально подходит для установки, в том числе на промышленных роботах и медицинском оборудовании [4].

Компактный модуль NVIDIA Jetson Xavier NX обеспечивает до 21 трлн операций в секунду для вычислений ИИ в периферийных устройствах. Вычислительное устройство обеспечивает параллельную работу нескольких нейронных сетей и обработку данных с нескольких датчиков высокого разрешения, что необходимо для систем ИИ. Jetson Xavier NX представляет собой готовое решение и поддерживает все популярные фреймворки ИИ [5].

NVIDIA Jetson AGX Xavier – это компактный мощный модуль гарантирует аппаратное ускорение для всего конвейера ИИ и высокую скорость передачи данных. С помощью Jetson AGX Xavier обеспечивает устойчивость к широкому диапазону температур, ударов и вибраций, а также новые возможности функциональной безопасности, которые хотят создавать продукты промышленного класса и/или сертифицированные по технической безопасности [6].

NVIDIA CUDA повышает производительность центрального процессора, создавая вычисления с ускорением графического процессора, которые выполняются быстрее, чем при традиционной обработке на центральном процессоре.

Помимо стандартных вычислений с одинарной точностью, видеокарты NVIDIA серий Pascal и Volta также поддерживают вычисления с низкой точностью FP16 и INT8.

Каждое тензорное ядро предоставляет матричный обрабатывающий массив $4\times4\times4$, который выполняет операцию $D=A\cdot B+C$, где A, B, C и D представляют собой матрицы 4×4, как показано на рис. 1. Входы матричного умножения A и B являются матрицами FP16, тогда как матрицы C и D накопления могут быть матрицами FP16 или FP32.

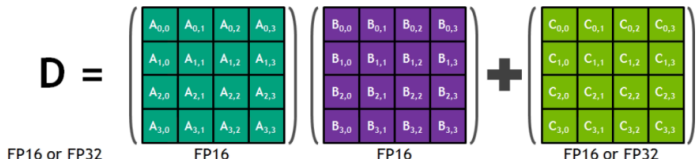

Рис. 1. Умножение и накопление матрицы тензорного ядра 4x4x4 [7]

Тензорные ядра работают с входными данными FP16 с накоплением FP32. Умножение FP16 приводит к результату полной точности, который накапливается в операциях FP32 с другими произведениями в данном скалярном произведении для умножения матрицы $4 \times 4 \times 4$, как показано на рис. 2.

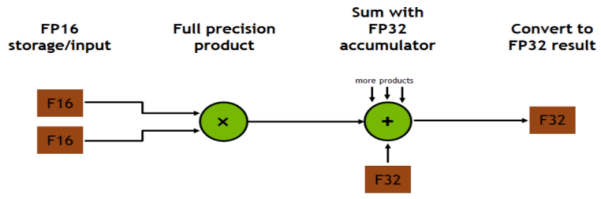

Рис. 2. Работа тензорного ядра [7]

Кроме аппаратных решений NVIDIA CUDA предоставляет спектр библиотек NVIDIA JetPack SDK: CUDA, cuDNN, TensorRT для запуска глубоких нейронных сетей. Библиотека cuDNN – это библиотека примитивов с ускорением на GPU для глубоких нейронных сетей, которая оптимизирует производительность нейронных сетей на графических процессорах NVIDIA. Библиотека TensorRT обеспечивает логический вывод с малой задержкой и высокой пропускной способностью и оптимизирует выполнение операций на различных семействах графических процессоров.

В настоящее время реализация СНС на периферийные вычислительные устройства массового производства на основе GPU имеют широкое применение в рамках мирового масштаба из-за своих технических характеристик, которые описаны в следующей главе.

2. Применение детекционных сверточных нейронных сетей на встраиваемых устройствах с ограниченным вычислительным ресурсом NVIDIA Jetson. В данном разделе рассмотрена реализация моделей детекционных сверточных нейронных сетей архитектур YOLOv3 [8, 9, 10] и ее укороченной модификации YOLOv3-TINY на встраиваемых вычислительных устройствах NVIDIA Jetson Nano B01, NVIDIA Jetson TX2 (8GB) и NVIDIA Jetson AGX Xavier [11], а также методы оптимизации нейросетевых алгоритмов по таким критериям как потребляемая мощность, время выполнения прямого прохода нейронной сети [12] и обеспечиваемая точность детекционного алгоритма по интегральному критерию Mean Average Precision (mAP) [13].

В основе модели нейронной сети YOLOv3 лежит принцип извлечения информации об объектах разного масштаба, который получил название Feature Pyramid Networks (FPN) [14, 15]. Модель нейронной сети реализована на backbone darknet.53 [8] с тремя регрессионными (детекторами) выходами, которые предсказывают параметры расположения объектов малых, средних и больших размеров. Модель НС YOLOv3-TINY реализована по тому же принципу, что и модель YOLOv3 только имеет backbone меньшей глубиной с двумя регрессионными выходами. Решением детекционного алгоритма может быть большое количество перекрывающихся объектов на изображении.

Для фильтрации решений используется алгоритм Non Maximum Suppression (NMS) [16, 17]. Это алгоритм выбирает один объект из множества перекрывающихся объектов по выдвинутому критерию. В большинстве случаев используют некую меру перекрытия, например, Intersection Over Union (IOU) (рис. 3) [18]. От типа реализации алгоритма NMS, архитектуры нейронной сети, весовых коэффициентов и т.д. время работы алгоритма может значительно варьироваться, поэтому в данном исследовании измерение времени прямого прохода нейронной сети не учитывает этот алгоритм.

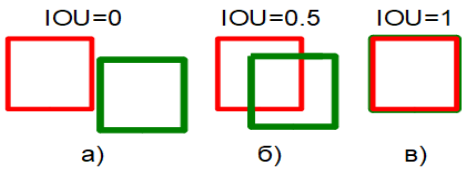

Рис. 3. Иллюстрация предсказанного и истинного обрамляющих прямоугольников для различных значений IOU а) 0 б) 0.5 в) 1

3. Вычислительный эксперимент. Оценки времени выполнения прямого прохода и точности обнаружения [19, 20] производились с применением программного обеспечения (ПО) NIVDIA JetPack 4.4.1 включающего в себя: CUDA 10.2, cuDNN 8.0, TensorRT 7.1.3 и библиотеки компьютерного зрения OpenCV 4.1.1. Все вычисления производились в режиме максимально допустимой вычислительной мощности (режим jetson clocks). В таблице 1 приведены показатели потребляемой мощности исследуемых вычислительных устройств для данного режима.

Обучение и оценка исследуемых характеристик нейросетевых алгоритмов производилась на наборе изображений для обнаружения объектов PASCAL VOC 2007 и PASCAL VOC 2012 [21], включающих в себя объекты 20 классов. Обучающая выборка состоит из 15000 изображений, а тестовая из 5000. Обучение и оценки исследуемых характеристик производились на изображениях размерностью 416 x 416 x 3 элементов яркости.

Оценка времени прямого прохода нейронной сети производилась на одном изображении для разных режимов вычислительной точности FP32, FP16, INT8. Следует отметить, что из исследуемых встраиваемых устройств, вычислительную точность INT8 поддерживает только на устройстве NVIDIA Jetson AGX Xavier. Результаты измерения времени прямого прохода исследуемых моделей нейронных сетей приведены в табл. 1.

Оценки характеристики точности обнаружения нейросетевых алгоритмов производилось на основе интегрального критерия mAP, которая рассчитывается как среднее между средней точностью обнаружения Average Precision (AP) каждого класса. В табл. 1 приведены результаты оценок точности обнаружения детекционных нейросетевых алгоритмов YOLOv3 и YOLOv3-TINY на основе критерия mAP.

В режиме реального времени прямой проход архитектуры YOLOv3 возможен только на вычислительном устройстве NVIDIA Jetson AGX Xavier для всех точностей. Архитектуру YOLOv3-TINY можно запустить в режиме реального времени на всех устройствах линейки NVIDIA Jetson для точности FP16. Вычислитель NVIDIA Jetson AGX Xavier обеспечивает меньшее время прямого прохода для исследуемых архитектур нейросетевых алгоритмов, однако потребляет большую мощность и находится в более высоком ценовом диапазоне.

Таблица 1

Результаты экспериментального исследования

Наименование вычислительного устройства	Модель нейронной сети	Точность FP32, FP16, INT8	Потребляемая мощность, Вт	TOPS	Время выполнения прямого прохода, мс	mAP, %
Jetson Nano B01 (4GB)	YoloV3 416	FP32	10	65.86	Превышение отведенной памяти	-
		FP16			201	71.54
		INT8			Нет поддержки	Нет поддержки

	YoloV3-tiny 416	FP32		5.477	45	51
		FP16			29.7	50.8
		INT8			Нет поддержки	Нет поддержки
Jetson TX2 (8GB)	YoloV3 416	FP32	15	65.86	148	71.55
		FP16			95	71.54
		INT8			Нет поддержки	Нет поддержки
	YoloV3-tiny 416	FP32		5.477	18.9	51
		FP16			13.7	50.8
		INT8			Нет поддержки	Нет поддержки
Jetson AGX Xavier	YoloV3 416	FP32	30	65.86	51	71.55
		FP16			17.3	71.54
		INT8			12.1	71.13
	YoloV3-tiny 416	FP32		5.477	6.42	51
		FP16			3.49	50.8
		INT8			2.71	49.28

Для решения задач на встраиваемых вычислительных устройствах нейросетевыми алгоритмами, целесообразно найти оптимум между предоставляемым и необходимым вычислительным ресурсом для работы нейросетевого алгоритма в режиме реального времени.

Заключение. На вычислительных устройствах NVIDIA Jetson Nano B01, NVIDIA Jetson TX2, NVIDIA Jetson AGX Xavier реализованы модели детекционных сверточных нейронных сетей YOLOv3 и YOLOv3-TINY, обученные на тренировочном наборе изображений. Рассмотрены программные и аппаратные решения оптимизации вычислений, предлагаемые компанией NVIDIA.

Произведено исследование влияния перехода к вычислениям с пониженной точностью FP16 и INT8 на время прямого прохода исследуемых архитектур нейронных сетей. Переход к вычислениям с пониженной точностью увеличивает скорость работы нейросетевых алгоритмов до 4.2 раза, что позволяет реализовать глубокие сверточные нейронные сети в режиме реального времена на периферийных вычислительных устройствах с малым форм–фактором и ограниченным вычислительным ресурсом. Чем грубее точность вычислений, тем быстрее работает алгоритм. Однако влияния вычисления с пониженной точностью приводит к снижению критерия Mean Average Precision до 1.7 %, характеризующего интегральную точность обнаружения объектов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. ГОСТ Р 59277—2020. Системы искусственного интеллекта. Классификация систем искусственного интеллекта.
2. Описание линейки NVIDIA Jetson // Официальный сайт NVIDIA. – URL: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems (дата обращения: 14.11.21).
3. Описание NVIDIA Jetson Nano // Официальный сайт NVIDIA. – URL: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems/jetson-nano (дата обращения: 15.11.21).
4. Описание NVIDIA Jetson TX2 // Официальный сайт NVIDIA. – URL: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems/jetson-tx2 (дата обращения: 16.11.21).
5. Описание NVIDIA Jetson Xavier NX // Официальный сайт NVIDIA. – URL: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems/jetson-xavier-nx (дата обращения: 17.11.21).

6. Описание NVIDIA Jetson AGX Xavier // Официальный сайт NVIDIA. – URL: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems/jetson-agx-xavier (дата обращения: 15.11.21).
7. Программирование тензорных ядер в CUDA // Официальный сайт NVIDIA. – URL: https://developer.nvidia.com/blog/programming-tensor-cores-cuda-9 (дата обращения: 16.11.21).
8. *Redmon J., Farhadi A.* YOLOv3: An Incremental Improvement // arXiv, 2018. Available at: https://arxiv.org/abs/1804.02767v1.
9. *Redmon J., Divvala S., Girshick R., and Farhadi A.* You only look once: Unified, real-time object detection // IEEE conference on computer vision and pattern recognition, 2016.
10. *Redmon J. and Farhadi A.* Yolov3: An incremental improvement // arXiv preprint arXiv:1804.02767, 2018.
11. *Elias Stein, Siyu Liu, John Sun* Real-Time Object Detection on an Edge Device (Final Report) // CS230: Deep Learning, 2019.
12. *Sazli Murat H.* A brief review of feed-forward neural networks // Ankara University, Faculty of Engineering, Department of Electronics Engineering.
13. *Van Etten A.* Satellite imagery multiscale rapid detection with windowed networks // IEEE Winter Conference on Applications of Computer Vision, 2019.
14. *Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie* Feature Pyramid Networks for Object Detection // IEEE Conference on Computer Vision and Pattern Recognition, 2017.
15. *Girshick R., Donahue J., Darrell T., and Malik J.* Rich feature hierarchies for accurate object detection and semantic segmentation // IEEE conference on computer vision and pattern recognition, 2014.
16. *He K., Zhang X., Ren S., and Sun J.* Identity mappings in deep residual networks // European Conference on Computer Vision, 2016.
17. *Jan Hosang, Rodrigo Benenson, Bernt Schiele* Learning Non-maximum Suppression // IEEE Conference on Computer Vision and Pattern Recognition, 2017.
18. *Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, Silvio Savarese*, Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression // arXiv, 2019. Available at: https://arxiv.org/abs/1902.09630.
19. *Prakhar Ganesh, Yao Chen, Yin Yang, Deming Chen, Marianne Winslett* YOLO-ReT: Towards High Accuracy Real-time Object Detection on Edge GPUs // Computer Vision and Pattern Recognition, 2021.
20. *Huang J., Rathod V., Sun C., Zhu M., Korattikara A., Fathi A., Fischer I., Wojna Z., Song Y., Guadarrama S., et al*. Speed/accuracy trade-offs for modern convolutional object detectors // IEEE Conference on Computer Vision and Pattern Recognition, 2017.
21. *Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn, and Andrew Zisserman.* The pascal visual object classes (voc) challenge // International Journal of Computer Vision, 2010.

REFERENCES

1. GOST R 59277—2020. Sistemy iskusstvennogo intellekta. Klassifikatsiya sistem iskusstvennogo intellekta [GOST R 59277—2020. Artificial intelligence systems. Classification of artificial intelligence systems].
2. Opisanie lineyki NVIDIA Jetson [Description of the NVIDIA Jetson line], *Ofitsial'nyy sayt NVIDIA* [Official site of NVIDIA]. Available at: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems (accessed 14 November 2021).
3. Opisanie NVIDIA Jetson Nano [Description of NVIDIA Jetson Nano], *Ofitsial'nyy sayt NVIDIA* [Official site of NVIDIA]. Available at: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems/jetson-nano (accessed 15 November 2021).
4. Opisanie NVIDIA Jetson TX2 [Description of NVIDIA Jetson TX2], *Ofitsial'nyy sayt NVIDIA* [Official site of NVIDIA]. Available at: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems/jetson-tx2 (accessed 16 November 2021).
5. Opisanie NVIDIA Jetson Xavier NX [Description of NVIDIA Jetson Xavier NX], *Ofitsial'nyy sayt NVIDIA* [Official site of NVIDIA]. Available at: URL: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems/jetson-xavier-nx (accessed 17 November 2021).

6. Opisanie NVIDIA Jetson AGX Xavier [Description of NVIDIA Jetson AGX Xavier] *Ofitsial'nyy sayt NVIDIA* [Official site of NVIDIA]. Available at: https://www.nvidia.com/ru-ru/autonomous-machines/embedded-systems/jetson-agx-xavier (accessed 15 November 2021).
7. Programmirovanie tenzornykh yader v CUDA [Programming tensor kernels in CUDA] *Ofitsial'nyy sayt NVIDIA* [Official site of NVIDIA]. Available at: https://developer.nvidia.com/blog/programming-tensor-cores-cuda-9 (accessed 16 November 2021).
8. *Redmon J., Farhadi A.* YOLOv3: An Incremental Improvement, *arXiv, 2018*. Available at: https://arxiv.org/abs/1804.02767v1.
9. *Redmon J., Divvala S., Girshick R., and Farhadi A.* You only look once: Unified, real-time object detection, *IEEE conference on computer vision and pattern recognition, 2016*.
10. *Redmon J. and Farhadi A.* Yolov3: An incremental improvement, *arXiv preprint arXiv:1804.02767, 2018.*
11. *Elias Stein, Siyu Liu, John Sun* Real-Time Object Detection on an Edge Device (Final Report), *CS230: Deep Learning, 2019*.
12. *Sazli Murat H.* A brief review of feed-forward neural networks, *Ankara University, Faculty of Engineering, Department of Electronics Engineering*.
13. *Van Etten A.* Satellite imagery multiscale rapid detection with windowed networks, *IEEE Winter Conference on Applications of Computer Vision, 2019.*
14. *Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie* Feature Pyramid Networks for Object Detection, *IEEE Conference on Computer Vision and Pattern Recognition, 2017*.
15. *Girshick R., Donahue J., Darrell T., and Malik J.* Rich feature hierarchies for accurate object detection and semantic segmentation, *IEEE conference on computer vision and pattern recognition, 2014*.
16. *He K., Zhang X., Ren S., and Sun J.* Identity mappings in deep residual networks, *European Conference on Computer Vision, 2016*.
17. *Jan Hosang, Rodrigo Benenson, Bernt Schiele* Learning Non-maximum Suppression, *IEEE Conference on Computer Vision and Pattern Recognition, 2017.*
18. *Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, Silvio Savarese*, Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression, *arXiv, 2019*. Available at: https://arxiv.org/abs/1902.09630.
19. *Prakhar Ganesh, Yao Chen, Yin Yang, Deming Chen, Marianne Winslett* YOLO-ReT: Towards High Accuracy Real-time Object Detection on Edge GPUs, *Computer Vision and Pattern Recognition, 2021*.
20. *Huang J., Rathod V., Sun C., Zhu M., Korattikara A., Fathi A., Fischer I., Wojna Z., Song Y., Guadarrama S., et al*. Speed/accuracy trade-offs for modern convolutional object detectors, *IEEE Conference on Computer Vision and Pattern Recognition, 2017*.
21. *Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn, and Andrew Zisserman.* The pascal visual object classes (voc) challenge, *International Journal of Computer Vision, 2010.*

Статью рекомендовал к опубликованию д.т.н., профессор В.В. Курейчик.

Ковалев Владислав Владимирович – Южный федеральный университет; e-mail: vlad.kovalev94@mail.ru; г. Таганрог, Россия; тел.: +79525864492; кафедра вычислительной техники; аспирант.

Сергеев Николай Евгеньевич – e-mail: nesergeev@sfedu.ru; тел.: +79281742585; кафедра вычислительной техники; д.т.н.; профессор.

Kovalev Vladislav Vladimirovich – Southern Federal University**;** e-mail: vlad.kovalev94@mail.ru; Taganrog, Russia; phone: +79525864492; the department of computer science; post-graduate student.

Sergeev Nikolay Evgenievich – e-mail: nesergeev@sfedu.ru; phone: +79281742585; the department of computer science; dr. of eng. sc.; professor.