

Оценка эффективности сверточных нейронных сетей в задаче автономного пилотирования

Т. В. Азарнова¹, Н. Г. Аснина^{2*}, М. А. Куприн²

¹ ФГБОУ ВО «Воронежский государственный университет», г. Воронеж, Российская Федерация
Адрес: 394018, Российской Федерации, г. Воронеж, Университетская площадь, д. 1

² ФГБОУ ВО «Воронежский государственный технический университет», г. Воронеж, Российской Федерации
Адрес: 394006, Российской Федерации, г. Воронеж, ул. 20-летия Октября, д. 84

* andrey050569@yandex.ru

Аннотация

Введение. Данное исследование посвящено разработке системы автономного перемещения беспилотной тележки GreenCamel АгроМул RC2400 по территории предприятия. Ключевой задачей является создание системы машинного зрения, способной в реальном времени распознавать препятствия, такие как дорожные конусы, и предпринимать действия для их объезда.

Материалы и методы. В качестве решения для задачи детекции объектов был выбран предобученный алгоритм YOLOv11, являющийся последней версией популярного семейства детекторов объектов в реальном времени. YOLOv11 обладает улучшенной архитектурой и методами обучения, что делает его эффективным инструментом для решения широкого спектра задач компьютерного зрения. Для обучения модели использовались данные из набора Microsoft COCO, дополненные датасетом дорожных конусов из открытого источника roboflow. Из 80 классов, представленных в Microsoft COCO, были выбраны 9 наиболее релевантных для задачи, а также добавлен класс «traffic cone». В ходе обучения модель YOLOv11 продемонстрировала высокую точность и эффективность, достигнув значения MAP (среднее значение показателя средней точности по всем классам) в 91.2%, Precision (точность) – 90.1% и Recall (полнота) – 86.5%.

Результаты исследования. В результате исследования было подтверждено, что YOLOv11 является эффективным инструментом для обнаружения дорожных конусов. Полученные результаты демонстрируют возможность использования алгоритма для создания системы автономного перемещения беспилотной тележки GreenCamel АгроМул RC2400.

Обсуждение и заключение. В качестве ограничений использования алгоритма можно отметить, что: сбор и разметка данных для задач детекции являются трудоемкими процессами, требующими качественной подготовки аннотаций; проблема дисбаланса классов может возникнуть при добавлении специализированных малочисленных классов, как в случае с «traffic cone». Несмотря на эти ограничения, YOLOv11 является эффективным инструментом обнаружения объектов окружения и получения данных, которые могут быть использованы для решения других задач, например определения расстояния до объекта по его площади на изображении.

Ключевые слова: сверточные нейронные сети, YOLOv11, машинное зрение, детекция объектов, Python, автономное перемещение

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Для цитирования: Азарнова Т. В., Аснина Н. Г., Куприн М. А. Оценка эффективности сверточных нейронных сетей в задаче автономного пилотирования // Современные информационные технологии и ИТ-образование. 2024. Т. 20, № 3. С. 638-644. <https://doi.org/10.25559/SITITO.020.202403.638-644>

© Азарнова Т. В., Аснина Н. Г., Куприн М. А., 2024



Контент доступен под лицензией Creative Commons Attribution 4.0 License.
The content is available under Creative Commons Attribution 4.0 License.



Evaluation of the Effectiveness of Convolutional Neural Networks in the Task of Autonomous Piloting

T. V. Azarnova^a, N. G. Asnina^{b*}, M. A. Kuprin^b

^a Voronezh State University, Voronezh, Russian Federation

Address: 1 University square, Voronezh 394018, Russian Federation

^b Voronezh State Technical University, Voronezh, Russian Federation

Address: 20 let Oktyabrya St., build. 84, Voronezh 394006, Russian Federation

* andrey050569@yandex.ru

Abstract

Introduction. This study is devoted to the development of a system for autonomous movement of the GreenCamel AgroMul RC2400 unmanned trolley across the territory of an enterprise. The key task is to create a machine vision system capable of real-time recognition of obstacles, such as traffic cones, and taking actions to bypass them.

Materials and Methods. As a solution for the task of object detection, the pre-trained YOLOv11 algorithm, which is the latest version of a popular family of real-time object detectors, was chosen. YOLOv11 has an improved architecture and training methods, which makes it an effective tool for solving a wide range of computer vision problems. To train the model, data from the Microsoft COCO dataset was used, supplemented by a dataset of traffic cones from the open source roboflow. Of the 80 classes presented in Microsoft COCO, the 9 most relevant to the task were selected, and the "traffic cone" class was also added. During the training, the YOLOv11 model demonstrated high accuracy and efficiency, achieving a MAP (average value of the average accuracy indicator for all classes) value of 91.2%, Precision (accuracy) – 90.1% and Recall (recall) – 86.5%.

Results. As a result of the study, it was confirmed that YOLOv11 is an effective tool for detecting traffic cones. The results demonstrate the possibility of using the algorithm to create an autonomous movement system for the GreenCamel AgroMul RC2400 unmanned cart.

Discussion and Conclusion. As limitations of using the algorithm, it can be noted that: collecting and marking data for detection tasks are labor-intensive processes that require high-quality preparation of annotations; the problem of class imbalance may arise when adding specialized small classes, as in the case of "traffic cone". Despite these limitations, YOLOv11 is an effective tool for detecting environmental objects and obtaining data that can be used to solve other problems, such as determining the distance to an object by its area in the image.

Keywords: convolutional neural networks, YOLOv11, machine vision, object detection, Python, autonomous movement

Conflict of interests: The authors declares no conflict of interest.

For citation: Azarnova T.V., Asnina N.G., Kuprin M.A. Evaluation of the Effectiveness of Convolutional Neural Networks in the Task of Autonomous Piloting. *Modern Information Technologies and IT-Education*. 2024;20(3):638-644. <https://doi.org/10.25559/SITITO.020.202403.638-644>



1. Введение

Разработка автопилотируемой техники является актуальной и важной задачей. На рынке уже существуют успешные решения на основе машинного зрения, такие как Tesla Model S Plaid, являющийся автомобилем с автопилотом высокого уровня, или PCM Агротроник Пилот 2.0 и Cognitive Agro Pilot, являющиеся автопилотами сельскохозяйственной техники. Несмотря на существование качественных решений в ряде областей, существует еще множество нерешенных задач. Одной из таких задач является конечная цель нашего исследования, а именно разработка системы автономного перемещения беспилотной тележки GreenCamel АгроМул RC2400 по территории предприятия. Система должна в реальном времени распознавать ключевые объекты на изображении, полученном с камеры машинного зрения, и предпринимать действия для предотвращения столкновений и объезда препятствий. Предварительно система должна распознавать следующие объекты: человек, автомобиль, велосипед, автобус, грузовик, птица, собака, кошка, дорожный конус, поддон, металлические трубы, дефекты дорожного покрытия.

Из этой комплексной задачи можно выделить подзадачу машинного зрения – детекцию объектов, являющуюся актуальной и распространенной задачей в области компьютерного зрения. Проблеме детекции объектов посвящен целый ряд работ российских и зарубежных исследователей [1-15]. В работе Kiela D. [1] представлена контролируемая модель мульти-модального битрансформатора, которая объединяет информацию из кодировщиков текста и изображений и достигает самых современных показателей производительности в различных тестовых задачах мульти-модальной классификации, превосходя сильные базовые показатели, в том числе на жестких тестовых наборах, специально разработанных для измерения мульти-модальной производительности. В статье Маркевича М. В. [2] показана методика нахождения ключевых слов, наиболее близко подходящих для описания изображения. Для этого оценивается близость между вектором изображения и вектором слова. Те вектора слов, которые оказываются наиболее близки к вектору изображения и используются в качестве ключевых слов. В статье проведено сравнение с обычной классификацией на 1000 классах изображений датасета ImageNet. Работа Diamos G., Sengupta S., Catanzaro B., Chrzanowski M., Coates, A., Elsen E., Engel J., Hannun A., Satheesh S. [4] посвящена вопросам повышения эффективности решения задач детекции за счет применения методики сопоставления параметров глубоких рекуррентных нейронных сетей (RNN) с возможностями графических процессоров, и достижения существенно более высокой вычислительной производительности при небольших размерах мини-пакетов, чем при прямой реализации RNN на основе матричного умножения. Предлагаемый подход реализуется за счет использования постоянных вычислительных ядер, и инвертированной иерархии памяти графического процессора для повторного использования весов сети на протяжении нескольких временных шагов. Реализация поддерживает производительность 2,8 т/с при размере мини-пакета 4 на графическом процессоре NVIDIA TitanX. Это обеспечивает 16-кратное сокращение объема памяти активации, позволяет обучать модели с использованием в 12 раз большего количе-

ства параметров на том же оборудовании. В статье Goyal P., Dollar P., Girshick R., Noordhuis P., Wesolowski L., Kyrola A., Tulloch A., Jia Y., He K. [5] рассматривается методика повышения эффективности решения задач машинного зрения за счет разделения мини-пакетов SGD на пул параллельных рабочих процессов. Применяется правило линейного масштабирования без гиперпараметров для корректировки скорости обучения в зависимости от размера мини-пакета и разрабатывается новая схема обработки, которая преодолевает проблемы оптимизации на ранних этапах обучения. Предложенная система на основе Caffe2 обучает ResNet-50 с размером мини-пакета 8192 на 256 графических процессорах за один час, обеспечивая при этом небольшую точность мини-пакета и достигает эффективности масштабирования 90% при переходе от 8 до 256 графических процессоров. Результаты позволяют с высокой эффективностью обучать модели визуального распознавания на данных интернет-масштаба. Работа Radiuk P.M. [9] посвящена проблеме повышения производительности сверточных нейронных сетей. Исследуется влияние размера пакета обучающего набора на производительность, используются различные по сложности наборы данных MNIST и CIFAR-10. Подтверждается предположение о зависимости точности распознавания от значения размера пакета и не подтверждается предположение о влиянии типа значения размера пакета на производительность CNN. В статье Mishkin D., Sergievskiy N., Matas J. [10] изучается влияние ряда последних достижений в архитектуре и методах обучения сверточных нейронных сетей (CNN) на проблему категоризации объектов (ILSVRC). В ходе оценки проверяется влияние следующих параметров архитектуры: нелинейность (ReLU, ELU, tanh), варианты пула (стохастический, максимальный, средний, смешанный), ширина сети, конструкция классификатора (сверточный, полностью – connected, SPP), предварительная обработка изображений, скорость обучения, размер пакета, чистота данных и т. д. Прирост производительности предлагаемых модификаций сначала тестируется индивидуально, а затем в сочетании. Авторы показывают, что использование изображений размером 128×128 пикселей достаточно, чтобы сделать качественные выводы об оптимальной структуре сети, справедливые для полноразмерных сетей Caffe и VGG. Результаты получаются на порядок быстрее, чем при использовании стандартных изображений размером 224 пикселя. В работе He K., Zhang X., Ren S., Sun J. [12] анализируются вопросы снижения сложности обучения глубоких нейронных сетей. Используется обучение функций остатка со ссылкой на входные данные слоя вместо обучения неиспользуемых функций. Результаты показывают, что остаточные сети легче оптимизировать и они могут повысить точность за счет значительно большей глубины. В наборе данных ImageNet оцениваются остаточные сети с глубиной до 152 слоев – в 8 раз глубже, чем сети VGG. Ансамбль этих остаточных сетей достигает ошибки 3,57% на тестовом наборе ImageNet. Глубина представлений имеет решающее значение для многих задач визуального распознавания. В статье Simonyan K., Zisserman A. [14] исследуется влияние глубины сверточных сетей на их точность в условиях крупномасштабного распознавания изображений. Проводится тщательная оценка сетей возрастающей глубины с использованием архитектуры с очень маленькими (3x3) сверточными фильтрами,



которая показывает, что значительное улучшение по сравнению с конфигурациями предшествующего уровня техники может быть достигнуто за счет увеличения глубины до 16-19 весовых слоев. Несмотря на большое количество работ в области машинного зрения, применения сверточных нейронных сетей, глубокого обучения, развития инструментов детектирования изображений, актуальными являются и исследования в области теоретических аспектов повышения производительности используемых подходов, так и работы, посвященные решению конкретных прикладных задач.

В представленной в данной статье работе оценивается эффективность использования сверточных нейронных сетей для распознавания препятствий. Исследование базируется на использовании сети YOLO (*You Only Look Once*).

2. Постановка задачи

Целью данного исследования является обучение нейронной сети для задачи детекции дорожных конусов, и оценка ее точности и эффективности по двум метрикам: точности определения класса, точности построения границ объекта на изображении. Из перечисленных ранее объектов выбраны именно дорожные конусы, потому что они будут использоваться для разметки маршрута в ранних версиях проекта. В будущем система будет расширяться и сможет распознавать остальные препятствия, описанные во введении.

3. Материалы и методы

В качестве решения описанной подзадачи предложено использовать предобученную сверточную нейронную сеть YOLOv11. Это новейшая итерация в серии детекторов объектов реального времени Ultralytics YOLO. YOLOv11 представляет значительные улучшения в архитектуре и методах обучения, что делает его универсальным выбором для широкого спектра задач компьютерного зрения, таких как обнаружение объектов, сегментация, классификация, обнаружение и отслеживание ключевых точек. YOLO обучена на наборе данных Microsoft COCO (*Common Objects in Context*), который содержит 330 тысяч изображений, из которых 200 тысяч содержат аннотации для задач обнаружения, сегментации и создания субтитров. Набор данных включает в себя 80 категорий объектов, включая обычные объекты, такие как автомобили, велосипеды и животные, а также более специфические категории, такие как зонты, сумки и спортивный инвентарь. Самая большая версия YOLOv11 имеет емкость 56.9 миллионов параметров.

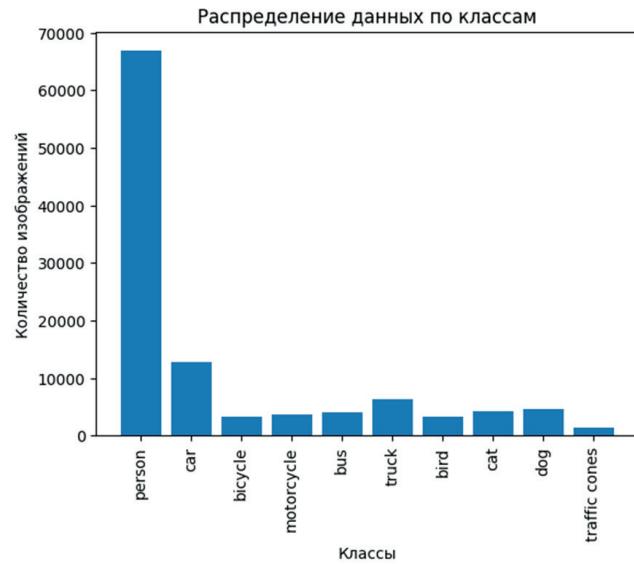
Для программной реализации используется язык программирования Python 3.11. Из основных библиотек стоит выделить открытую библиотеку OpenCV (*Open Source Computer Vision Library*) для работы с алгоритмами компьютерного зрения, машинного обучения и обработки изображений. В этом исследовании OpenCV использовалась для реализации обнаружения объектов в режиме реального времени.

4. Анализ и подготовка данных

Для обучения модели используются данные Microsoft COCO и датасет дорожных конусов, подобранный из свободного

источника roboflow. Roboflow – это платформа для разработки и внедрения моделей компьютерного зрения. Она предоставляет все необходимые инструменты для работы с визуальными данными, обучения моделей и их развертывания. Кроме того, на данной платформе имеется большое количество размеченных баз данных. Найденный на этой платформе набор данных, подходящий для нашего исследования, представляет набор из 1648 изображений, размеченных для задачи детекции объектов.

В контексте этой задачи нет необходимости в большинстве классов, присутствующих в Microsoft COCO, поэтому необходимо избавиться от чрезмерных данных. Из 80 классов оставлено 9: person, bicycle, car, motorcycle, bus, truck, bird, dog, cat. Кроме того, к перечисленным классам добавлен класс traffic cone из дополнительного датасета. Стоит обратить внимание на проблему неравномерного распределения данных по классам, которую хорошо видно на рисунке. 1. Введенный нами класс trafficcone значительно меньше любого из представленных в Microsoft COCO, это может привести к снижению качества модели, а в частности смещению в сторону предсказания больших классов. В результате модель может неверно классифицировать объекты меньшего класса.



Р и с. 1. Распределение данных по классам

F i g. 1. Distribution of data by classes

Источник: здесь и далее в статье все рисунки составлены авторами.

Source: Hereinafter in this article all figures were drawn up by the authors.

Для использования в обучении модели данные необходимо привести к формату YOLO. Для этого необходимо использовать ultralytics.data.converter.convert_coco. Эта функция преобразует аннотации набора данных COCO в формат аннотаций YOLO, подходящий для обучения моделей YOLO.

5. Обучение модели

Для обучения модели YOLO удобно создать конфигурационный файл формата YAML [15-17]. Это формат данных, который



используется для представления структурированных данных в простом формате. В отличие от таких форматов, как JSON или XML, YAML считается более простой и интуитивно понятной структурой. Он не является языком разметки, а предназначен для представления данных в структурированном виде. YAML используется для:

1. создания конфигурационных файлов приложений и сервисов;
2. сериализации и десериализации структур данных;
3. определения спецификаций и документации API;
4. описания конфигураций инфраструктуры.

Для текущей задачи он может выглядеть как показано на рисунке 2. В этом файле обязательно должны содержаться пути к папкам с изображениями, количество классов и их название.

```

1  train: train/images
2  val: valid/images
3  test: test/images
4
5  nc: 10
6  names: ['person',
7  'car',
8  'bicycle',
9  'motorcycle',
10 'bus',
11 'truck',
12 'bird',
13 'cat',
14 'dog',
15 'traffic cones']
16

```

Рис. 2. Содержание YAML файла

Fig. 2. Contents of YAML file

При рассмотрении результатов обучения нейронной сети сосредоточимся только на дорожных конусах, так как для остальных классов точность и эффективность модели YOLOv11 уже известна. На рисунке 3 изображены графики потерь для обучения для класса traffic con.

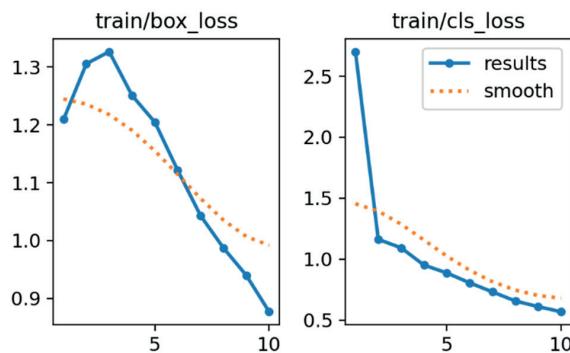


Рис. 3. Графики потерь для обучения

Fig. 3. Training Loss Graphs

Для реализации работы с камерой или видео файлом использовалась функция VideoCapture из библиотеки OpenCV. Она захватывает видео с камеры или файла. Он принимает только один аргумент: индекс используемой камеры или имя видеофайла, который нужно открыть.

6. Результаты исследования

В процессе обучения модели YOLOv11 мы получили значения MAP, Precision и Recall равными 91.2, 90.1 и 86.5 процентов соответственно. MAP показывает среднее значение показателя средней точности по всем классам модели, Precision определяет, насколько часто прогнозы вашей модели оказываются верными, а Recall показывает, какой процент соответствующих меток был успешно идентифицирован. Значение всех показателей превышает или приближается достаточно близко к 90 процентам, что является хорошим результатом. На рисунке 4 продемонстрирован результат обработки изображения обученной моделью.



Рис. 4. Обработанное моделью изображение

Fig. 4. Model processed image

Кроме того, на выходе модель предоставляет нам данные о координатах центра каждого объекта, высоту, ширину, название класса и вероятность принадлежности объекта к определенному моделью классу [18-22]. Выходная информация для рисунка 4 показана на рисунке 5.

```

{
  "predictions": [
    {
      "x": 347,
      "y": 220.5,
      "width": 34,
      "height": 71,
      "confidence": 0.809,
      "class": "trafficcone",
      "class_id": 0,
      "detection_id": "a3f38d5c-7ef1-432e-bf9b-633619efdb8"
    },
    {
      "x": 516,
      "y": 186.5,
      "width": 26,
      "height": 51,
      "confidence": 0.714,
      "class": "trafficcone",
      "class_id": 0,
      "detection_id": "19c7a81b-6bbf-4c34-b3a5-77f55d4d4732"
    }
  ]
}

```

Рис. 5. Вывод модели при предсказании

Fig. 5. Model output during prediction



7. Обсуждение и заключение

В ходе исследования к датасету Microsoft COCO был добавлен новый класс, на новых данных обучена нейронная сеть YOLOv11, подтверждена эффективность данной модели в распознавании всех используемых классов. Стоит отметить, что сбор и разметка данных для задач детекции очень трудоемкий процесс, требующий подготовки в формате YOLO качествен-

ных аннотаций к изображениям. Существует проблема дисбаланса классов, которая может проявиться при добавлении специализированных малочисленных классов. Несмотря на ограничения, YOLOv11 является качественным инструментом для обнаружения объектов окружения и получения данных о них, которые в будущем могут использоваться и для других задач, например определения расстояния до объекта по его площади на изображении [23-25].

References

- [1] Kiela D., et al. Supervised Multimodal Bitransformers for Classifying Images and Text. *arXiv:1909.02950*. 2019. <https://doi.org/10.48550/arXiv.1909.02950>
- [2] Markeev M.V. *Metodika avtomatizirovannoj razmetki izobrazhenij i nahozeniya klyuchevyh slov* [Methods of automated image markup and keyword finding]. *Mezhdunarodnyj zhurnal gumanitarnyh i estestvennyh nauk* = International Journal of Humanities and Natural Sciences. 2022;11-2(74):115-120. (In Russ., abstract in Eng.) <https://doi.org/10.24412/2500-1000-2022-11-2-115-120>
- [3] LeCun Y.A., Bottou L., Orr G.B., Müller KR. Efficient BackProp. In: Montavon G., Orr G.B., Müller KR. (eds.) *Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science*. Vol. 7700. Berlin, Heidelberg: Springer; 2012. p. 9-48. https://doi.org/10.1007/978-3-642-35289-8_3
- [4] Diamos G., Sengupta S., Catanzaro B., Chrzanowski M., Coates A., Elsen E., Engel J., Hannun A., Satheesh S. Persistent RNNs: Stashing recurrent weights on-chip. In: Balcan M.F., Weinberger K.Q. (eds.) *Proceedings of The 33rd International Conference on Machine Learning*. New York, New York, USA: PMLR; 2016. Vol. 48. p. 2024-2033. Available at: <https://proceedings.mlr.press/v48/diamos16.html> (accessed 26.05.2024).
- [5] Goyal P., Dollar P., Girshick R., Noordhuis P., Wesolowski L., Kyrola A., Tulloch A., Jia Y., He K. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. *arXiv:1706.02677*. 2017. p. 1-12. <https://doi.org/10.48550/arXiv.1706.02677>
- [6] Devarakonda A., Naumov M., Garland M. AdaBatch: Adaptive Batch Sizes for Training Deep Neural Networks. In: 6th International Conference on Learning Representations (ICLR 2018 Workshop Track). Vancouver, BC, Canada: Vancouver Convention Center; 2018. p. 1-4. Available at: <https://openreview.net/forum?id=SkytjjU8G> (accessed 26.05.2024).
- [7] Lisov A.A., Kulganatov A.Z., Panishev S.A. Using convolutional neural networks for acoustic-based emergency vehicle detection. *Modern Transportation Systems and Technologies*. 2023;9(1):95-107. (In Russ., abstract in Eng.) <https://doi.org/10.17816/ransyst20239195-107>
- [8] Avilov O., Rimbert S., Popov A., Bougrain L. Deep Learning Techniques to Improve Intraoperative Awareness Detection from Electroencephalographic Signals. In: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC). Montreal, QC, Canada: IEEE Computer Society; 2020. p. 142-145. <https://doi.org/10.1109/EMBC44109.2020.9176228>
- [9] Radiuk P.M. Impact of Training Set Batch Size on the Performance of Convolutional Neural Networks for Diverse Datasets. *Information Technology and Management Science*. 2017;20(1):20-24. <https://doi.org/10.1515/itms-2017-0003>
- [10] Mishkin D., Sergievskiy N., Matas J. Systematic evaluation of convolution neural network advances on the Imagenet. *Computer vision and image understanding*. 2017;161:11-19. <https://doi.org/10.1016/j.cviu.2017.05.007>
- [11] Bagby T., Rao K., Sim K.C. Efficient Implementation of Recurrent Neural Network Transducer in Tensorflow. In: 2018 IEEE Spoken Language Technology Workshop (SLT). Athens, Greece: IEEE Computer Society; 2018. p. 506-512. <https://doi.org/10.1109/SLT.2018.8639690>
- [12] He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE Computer Society; 2016. p. 770-778. <https://doi.org/10.1109/CVPR.2016.90>
- [13] Krizhevsky A. One weird trick for parallelizing convolutional neural networks. *arXiv:1404.5997v2*. 2014. <https://doi.org/10.48550/arXiv.1404.5997>
- [14] Simonyan K., Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In: 3rd International Conference on Learning Representations (ICLR 2015). *arXiv:1409.1556*. 2015. p. 1-15. <https://doi.org/10.48550/arXiv.1409.1556>
- [15] Takáč M., Bijral A., Richtárik P., Srebro N. Mini-Batch Primal and Dual Methods for SVMs. In: Dasgupta S., McAllester D. (eds.) *Proceedings of the 30th International Conference on Machine Learning (PMLR)*. 2013;28(3):1022-1030. Available at: <https://proceedings.mlr.press/v28/takac13.html> (accessed 26.05.2024).
- [16] Wilson D.R., Martinez T.R. The general inefficiency of batch training for gradient descent learning. *Neural networks*. 2003;16(10):14291451. [https://doi.org/10.1016/S0893-6080\(03\)00138-2](https://doi.org/10.1016/S0893-6080(03)00138-2)
- [17] Lin Z., Courbariaux M., Memisevic R., Bengio Y. Neural Networks with Few Multiplications. In: Bengio Y., LeCun Y. (eds.) 4th International Conference on Learning Representations, ICLR 2016. San Juan, Puerto Rico, May 2-4, 2016. Conference Track Proceedings. 2016. <https://doi.org/10.48550/arXiv.1510.03009>



- [18] Raissi M., Perdikaris P., Karniadakis G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*. 2019;378:686-707. <https://doi.org/10.1016/j.jcp.2018.10.045>
- [19] Cuomo S., et al. Scientific Machine Learning Through Physics-Informed Neural Networks: Where we are and What's Next. *Journal of Scientific Computing*. 2022;92(3):88. <https://doi.org/10.48550/arXiv.2201.05624>
- [20] Ulyanov S.V., Litvintseva L.V., Panfilov S.A. Design of self-organized intelligent control systems based on quantum fuzzy inference: intelligent system of systems engineering approach. In: 2005 IEEE International Conference on Systems, Man and Cybernetics, Waikoloa, HI, USA: IEEE Computer Society; 2005. Vol. 4. p. 3835-3840. <https://doi.org/10.1109/ICSMC.2005.1571744>
- [21] Liu Y., Feng S., Zhao Z., Ding E. Highly Efficient Human Action Recognition with Quantum Genetic Algorithm Optimized Support Vector Machine. *arXiv:1711.09511*. 2017. <https://doi.org/10.48550/arXiv.1711.09511>
- [22] Zhao C., Gao X. QDNN: deep neural networks with quantum layers. *Quantum Machine Intelligence*. 2021;3(1):15. <https://doi.org/10.1007/s42484-021-00046-w>
- [23] Vanchurin V. The World as a Neural Network. *Entropy*. 2020;22(11):1210. <https://doi.org/10.3390/e22111210>
- [24] Keysers D., Deselaers T., Gollan C., Ney H. Deformation Models for Image Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2007;29(8):1422-1435. <https://doi.org/10.1109/TPAMI.2007.1153>
- [25] Cireşan D.C., Meier U., Gambardella L.M., Schmidhuber J. Deep, Big, Simple Neural Nets for Handwritten Digit Recognition. *Neural Computation*. 2010;22(12):3207-3220. https://doi.org/10.1162/NECO_a_00052

Поступила 26.05.2024; одобрена после рецензирования 21.07.2024; принята к публикации 13.09.2024.

Submitted 26.05.2024; approved after reviewing 21.07.2024; accepted for publication 13.09.2024.

Об авторах:

Азарнова Татьяна Васильевна, заведующий кафедрой математических методов исследования операций факультета прикладной математики, информатики и механики, ФГБОУ ВО «Воронежский государственный университет» (394018, Российская Федерация, г. Воронеж, Университетская площадь, д. 1), доктор технических наук, профессор, ORCID: <https://orcid.org/0000-0001-6342-9355>

Аснина Наталья Георгиевна, заведующий кафедрой систем управления и информационных технологий в строительстве факультета информационных технологий и компьютерной безопасности, ФГБОУ ВО «Воронежский государственный технический университет» (394006, Российская Федерация, г. Воронеж, ул. 20-летия Октября, д. 84), кандидат технических наук, доцент, ORCID: <https://orcid.org/0000-0002-5857-108X>, andrey050569@yandex.ru

Куприн Михаил Андреевич, аспирант кафедры систем управления и информационных технологий в строительстве факультета информационных технологий и компьютерной, ФГБОУ ВО «Воронежский государственный технический университет» (394006, Российская Федерация, г. Воронеж, ул. 20-летия Октября, д. 84), ORCID: <https://orcid.org/0009-0009-7459-4837>, highman3@yandex.ru

Все авторы прочитали и одобрили окончательный вариант рукописи.

About the authors:

Tatyana V. Azarnova, Head of the Chair of Mathematical Methods of Operations Research, Applied Mathematics, Informatics and Mechanics Faculty, Voronezh State University (1 University square, Voronezh 394018, Russian Federation), Dr. Sci. (Tech.), Professor, ORCID: <https://orcid.org/0000-0001-6342-9355>

Natalia G. Asnina, Head of the Chair of Management Systems and Information Technologies in Construction, Faculty of Information Technology and Computer Safety, Voronezh State Technical University (20 let Oktyabrya St., build. 84, Voronezh 394006, Russian Federation), Cand. Sci. (Eng.), Associate Professor, ORCID: <https://orcid.org/0000-0002-5857-108X>, andrey050569@yandex.ru

Mikhail A. Kuprin, Postgraduate Student of the Chair of Management Systems and Information Technologies in Construction, Faculty of Information Technology and Computer Safety, Voronezh State Technical University (20 let Oktyabrya St., build. 84, Voronezh 394006, Russian Federation), ORCID: <https://orcid.org/0009-0009-7459-4837>, highman3@yandex.ru

All authors have read and approved the final manuscript.

