

Синтез архитектуры нейронной сети для распознавания образов морских судов

Д. И. Конарев¹, А. А. Гуламов¹✉

¹ ФГБОУ ВО «Юго-Западный государственный университет»
ул. 50 лет Октября 94, г. Курск 305040, Российская Федерация

✉ e-mail: profgulamov@mail.ru

Резюме

Цель исследования. В области инфокоммуникационного обеспечения судоходства Канала имени Москвы актуальной задачей является мониторинг судов с использованием камер видеонаблюдения, установленных на протяжении канала. Основной подзадачей является непосредственно распознавание судов на изображении или видео, для чего перспективно применение нейронной сети.

Методы. В работе рассмотрены различные архитектуры нейронной сети. Входными данными для сети являются изображения судов. Обучающая выборка использует набор данных CIFAR-10. Сеть построена и обучена с использованием библиотек машинного обучения Keras и TensorFlow.

Результаты. Описано применение свёрточных искусственных нейронных сетей для задач распознавания образов и преимущества такой архитектуры при работе с изображениями. Обоснован выбор языка Python для реализации нейронной сети и описаны основные применяемые библиотеки машинного обучения, такие, как TensorFlow и Keras. Проведён эксперимент по обучению свёрточных нейронных сетей с различной архитектурой на базе сервиса Google collaboratoty. Проведена оценка эффективности различных архитектур в процентном соотношении случаев правильного распознавания образов на тестовой выборке. Сделаны выводы о влиянии параметров свёрточной нейронной сети на проявление её эффективности.

Заключение. Сеть с одним свёрточным слоем в каждом каскаде показала недостаточные результаты, поэтому были рассмотрены трёхкаскадные свёрточные сети с двумя и тремя свёрточными слоями в каждом каскаде. Наибольшее влияние на точность распознавания образов оказало увеличение карты признаков. Наращивание числа каскадов оказало менее заметный эффект, а увеличение числа свёрточных слоёв в каждом каскаде не всегда приводит к повышению точности работы нейронной сети. В процессе исследования трёхкаскадная сеть с двумя свёрточными слоями в каждом каскаде и 128 картами признаков определена как оптимальная архитектура нейронной сети в рассматриваемых условиях. Проверка работоспособности части рассматриваемых архитектур на случайных изображениях судов подтвердила правильность выбора оптимальной архитектуры.

Ключевые слова: искусственные нейронные сети; свёрточная нейронная сеть; ядро свёртки; Keras; TensorFlow; Google collaboratoty; Cifar-10.

Конфликт интересов. Авторы декларируют отсутствие конфликта интересов, связанных с публикацией данной статьи.

Для цитирования: Конарев Д. И., Гуламов А. А. Синтез архитектуры нейронной сети для распознавания образов морских судов // Известия Юго-Западного государственного университета. 2020; 24(1): 130-143. <https://doi.org/10.21869/2223-1560-2020-24-1-130-143>.

Поступила в редакцию 14.10.2019

Подписана в печать 22.12.2019

Опубликована 21.02.2020

© Конарев Д. И., Гуламов А. А., 2020

Synthesis of Neural Network Architecture for Recognition of Sea-Going Ship Images

Dmitrii I. Konarev¹, Alisher A. Gulamov¹ ✉

¹ Southwest State University
50 Let Oktyabrya str. 94, Kursk 305040, Russian Federation

✉ e-mail: profgulamov@mail.ru

Резюме

Purpose of research. The current task is to monitor ships using video surveillance cameras installed along the canal. It is important for information communication support for navigation of the Moscow Canal. The main subtask is direct recognition of ships in an image or video. Implementation of a neural network is perspective.

Methods. Various neural network are described. Images of ships are an input data for the network. The learning sample uses CIFAR-10 dataset. The network is built and trained by using Keras and TensorFlow machine learning libraries.

Results. Implementation of curving artificial neural networks for problems of image recognition is described. Advantages of such architecture when working with images are also described. The selection of Python language for neural network implementation is justified. The main used libraries of machine learning, such as TensorFlow and Keras are described. An experiment has been conducted to train swirl neural networks with different architectures based on Google collaboratoty service. The effectiveness of different architectures was evaluated as a percentage of correct pattern recognition in the test sample. Conclusions have been drawn about parameters influence of screwing neural network on showing its effectiveness.

Conclusion. The network with a single curl layer in each cascade showed insufficient results, so three-stage curls with two and three curl layers in each cascade were used. Feature map extension has the greatest impact on the accuracy of image recognition. The increase in cascades' number has less noticeable effect and the increase in the number of screwdriver layers in each cascade does not always have an increase in the accuracy of the neural network. During the study, a three-frame network with two buckling layers in each cascade and 128 feature maps is defined as an optimal architecture of neural network under described conditions. operability checking of architecture's part under consideration on random images of ships confirmed the correctness of optimal architecture choosing.

Keywords: artificial neural networks; convolutional neural network; convolution kernel; Keras; TensorFlow; Google collaboratoty; Cifar-10.

Conflict of interest. The authors declare the absence of obvious and potential conflicts of interest related to the publication of this article.

For citation: Konarev D. I., Gulamov A. A. Synthesis of Neural Network Architecture for Recognition of Sea-Going Ship Images. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University*. 2020, 24(1): 130-143 (In Russ.). <https://doi.org/10.21869/2223-1560-2020-24-1-130-143>.

Received 14.10.2020

Accepted 22.12.2020

Published 21.02.2020

Введение

Исходя из цели разработки методов и системы программно-технических средств сбора, обработки, хранения, анализа параметров судоходных каналов, обеспечения навигации грузо- и пассажироперевозок в рамках информационно-телекоммуникационной системы мониторинга и управления судоходством Канала имени Москвы основной задачей является разработка комплекса клиентских программных средств сбора и анализа параметров судоходства [1, 2, 3]. Наряду с подсистемой хранения данных на базе технологии блокчейн необходимо организовать мониторинг судов.

В области инфокоммуникационного обеспечения мониторинга судоходства Канала имени Москвы основной подзадачей является непосредственно распознавание судов на изображении или видео, для чего перспективно применение нейронной сети. Применение видеосистемы обнаружения морских судов по триангуляционным решёткам представлено в работе [4]. Выделение контуров морских объектов на основе пирамидально-рекурсивного метода представления изображений исследуется в [5]. Выделение контуров изображений морских судов на основе построения видеосистемы искусственного зрения, автоматического обнаружения и распознавания водных (морских) транс-портных средств рассмотрено в [6]. Классифика-

ция морских судов с применением свёрточных нейронных сетей приведена в [7].

Свёрточная нейронная сеть – один из лучших алгоритмов по распознаванию и классификации изображений. По сравнению с полносвязной нейронной сетью (типа перцептрона) — гораздо меньшее количество настраиваемых весов, так как одно ядро весов используется целиком для всего изображения, вместо того, чтобы делать для каждого пикселя входного изображения свои персональные весовые коэффициенты [8, с. 115]. Это подталкивает нейросеть при обучении к обобщению демонстрируемой информации, а не попиксельному запоминанию каждой показанной картинке в мириадах весовых коэффициентов, как это делает перцептрон [9, с. 52].

При применении нейронных сетей для решения задач распознавания образов морских судов актуально использование языка Python [10, с. 15], библиотек TensorFlow и Keras. TensorFlow – это библиотека для работы с многомерными матрицами, тензорами, которые широко применяются при обучении нейронных сетей, и вычислений на графах потоков данных [11, с. 271]. Keras – это библиотека глубокого обучения, которую используют TensorFlow или Theano для выполнения эффективных вычислений [12, с. 177]. Особенность библиотеки Keras в том, что она позволяет на Python описывать нейронную сеть.

Материалы и методы

Полносвязные сети – самый простой тип нейронных сетей. Однако необходимо обучить большое количество весов даже для маленьких изображений и данные представляются в виде одномерного вектора или массива, таким образом теряется топологическая информация, важная для обработки изображений [13, с. 23]. Учитывается связь пикселей по горизонтали, но не по вертикали. Свёрточные нейронные сети лишены таких недостатков и преимущественно используются на практике для обработки изображений и видео. Свёрточные сети используют следующие принципы:

- локальное восприятие;
- разделяемые веса;
- уменьшение размерности.

В связи с этим, в процессе решения задачи для распознавания образов применялась свёрточная нейронная сеть с использованием алгоритма обучения с учителем [14, с. 396]. Операция свертки заключается в том, что рассматривается некоторая область изображения, например 3 на 3, значение интенсивности каждого пикселя в этом участке умножается на соответствующий элемент ядра свертки. Ядро свертки – матрица такого же размера, как рассматриваемый участок изображения. После этого все полученные элементы складываются [15, с. 592].

Рабочая сеть реализована на платформе Google collaboratory, а примерная структура простейшего её варианта представлена на рис. 1.

```
[ ] # Создание последовательной модели
model = Sequential()

# Сверточный слой
model.add(Conv2D(8, (3, 3), padding='same', input_shape=(32, 32, 3), activation='relu'))
# Слой подвыборки
model.add(MaxPooling2D(pool_size=(2, 2)))
# Слой регуляризации Dropout
model.add(Dropout(0.25))

# Слой преобразования данных из 2D представления в плоское
model.add(Flatten())
# Полносвязный слой для классификации
model.add(Dense(512, activation='relu'))
# Слой регуляризации Dropout
model.add(Dropout(0.5))
# Выходной полносвязный слой
model.add(Dense(nb_classes, activation='softmax'))
```

Рис. 1. Описание модели нейронной сети

Fig. 1. Description of the neural network model

Библиотека Keras уже содержит функции для загрузки набора данных CIFAR-10, 90% выборки используется как обучающая, а 10% – как проверочная [16, с. 161].

Входными данными нейронной сети является набор изображений в формате RGB. Разрешение изображений в наборе данных составляет 32 на 32 пикселя, что позволит нейронной сети рас-

познавать суда на изображениях небольшого разрешения, характерных для камер видеонаблюдения [17, с. 92]. На рис. 2 представлен пример входного изображения.

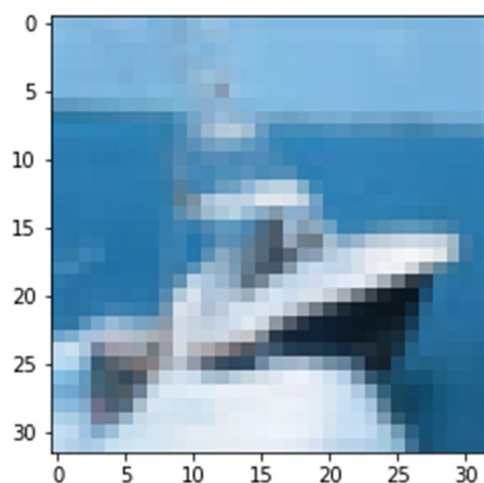


Рис. 2. Описание модели нейронной сети

Fig. 2. Description of the neural network model

Выходными данными нейронной сети является номер класса, к которому относится изображение. Если распознанный сетью класс изображения соответствует действительности, считаем ответ верным и указываем в таблице процент верных ответов при распознавании тестовой подвыборки. Этот параметр называется точностью – отношение верно угаданных объектов клас-

са ко всем объектам, которые мы определили как объекты класса.

Изменением параметров сети, таких, как число каскадов, число свёрточных слоёв в сети и количество карт признаков, проводилась оптимизация архитектуры сети. На рис. 3 представлена схема двухкаскадной сети с двумя слоями свёртки в каждом каскаде и 32 картами признаков. Для других сетей из эксперимента внешний вид схемы будет отличаться в зависимости от указанных выше параметров, но общая структура сети будет сохраняться. Так как параметры метода обучения установлены таким образом, что библиотека в начале каждой эпохи будет перемешивать данные для повышения качества обучения, то при каждой итерации обучения с одинаковыми параметрами результат работы сети будет незначительно отличаться, и для более точной оценки необходимо провести несколько итераций обучения [18, с. 370]. Вместе с тем каждая итерация обучения – длительный процесс, и чем глубже сеть, тем больше времени она занимает, поэтому необходимо выбрать минимальное достаточное число итераций [19, с. 80].

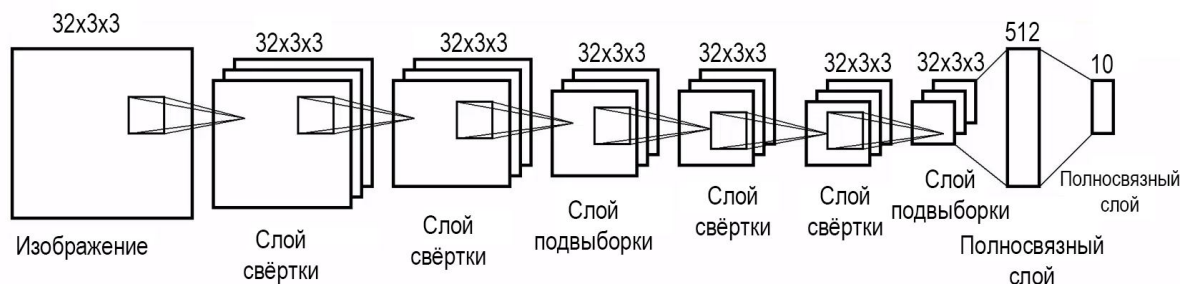


Рис. 3. Схема двухкаскадной сети с двумя слоями свёртки в каждом каскаде и 32 картами признаков

Fig. 3. Diagram of a two-stage network with two convolution layers in each stage and 32 feature cards

Результаты и их обсуждение

На основании полученных данных, предложено проводить 3 итерации обучения и оценивать сеть по среднему

значению. В табл. 1 представлены результаты работы нейронной сети из 2 каскадов с переменным размером карты признаков. В каждом каскаде имеется один свёрточный слой.

Таблица 1. Результаты работы разрешения изображений двухкаскадной нейронной сети с одним свёрточным слоем в каждом каскаде

Table 1. Image resolution results of a two-stage neural network with one convolutional layer in each stage

Число карт признаков	Первая итерация (%)	Вторая итерация (%)	Третья итерация (%)	Среднее значение (%)
8	63,2	65,53	63,67	64,13
16	71,16	70,56	71,96	71,23
24	71,47	69,83	70,33	70,54
32	73,79	73,99	74,64	74,14
40	75,24	75,43	73,98	74,88
48	75,52	75,47	75,27	75,42
64	76,08	75,15	75,61	75,61
128	74,57	75,41	75,22	75,07

Видим, что точность работы сети значительно увеличивается при увеличении числа карт до 32 признаков, дальнейшее наращивание числа карт признаков не даёт значительного прироста эффективности, а карта размером 128 признаков даже даёт снижение эф-

фективности работы, что может свидетельствовать о переобучении сети [20, с. 51].

Результаты для двухкаскадной сети с 2 свёрточными слоями приведены в табл. 2, с 3 свёрточными слоями – в табл. 3

Таблица 2. Результаты работы двухкаскадной нейронной сети с двумя свёрточными слоями в каждом каскаде

Table 2. The results of the work of a two-stage neural network with two convolutional layers in each cascade

Число карт признаков	Первая итерация (%)	Вторая итерация (%)	Третья итерация (%)	Среднее значение (%)
8	66,54	67,46	67,27	67,09
16	75,02	74,26	74,67	74,65
24	76,46	76,41	75,75	76,21
32	78,64	77,64	78,47	78,25
40	78,91	79,04	78,25	78,73
48	79,30	80,06	79,31	79,56
64	79,70	78,87	79,34	79,30
128	79,94	79,94	79,62	79,83

Таблица 3. Результаты работы двухкаскадной нейронной сети с тремя свёрточными слоями в каждом каскаде**Table 3.** The results of a two-stage neural network with three convolutional layers in each cascade

Число карт признаков	Первая итерация (%)	Вторая итерация (%)	Третья итерация (%)	Среднее значение (%)
8	65,38	65,77	66,74	65,96
16	73,05	74,14	74,82	74,00
24	76,68	75,56	76,56	76,27
32	78,19	78,67	79,49	78,78
40	79,21	78,71	79,52	79,15
48	79,64	79,82	80,72	80,06
64	81,19	80,83	80,44	80,82
128	81,67	81,49	81,60	81,59

В сравнении с предыдущей сетью средний прирост точности составляет всего 0,38%. Результаты добавления

ещё одного каскада, для двух свёрточных слоёв в каждом каскаде и приведены в табл. 4.

Таблица 4. Результаты работы трёхкаскадной нейронной сети с двумя свёрточными слоями в каждом каскаде**Table 4.** The results of a three-stage neural network with two convolutional layers in each cascade

Число карт признаков	Первая итерация (%)	Вторая итерация (%)	Третья итерация (%)	Среднее значение (%)
8	52,44	53,97	52,60	53,00
16	68,16	67,00	67,07	67,41
24	71,17	72,40	73,62	72,40
32	75,40	76,80	75,05	75,75
40	77,71	77,57	78,79	78,02
48	78,66	78,97	78,93	78,85
64	81,41	81,51	80,37	81,10
128	82,07	81,73	82,57	82,12

В сравнении с предыдущей сетью видим, что эффективность сети ухудшилась (в среднем на 1,83%).

Таким образом, эффективной архитектурой нейронной сети для рассматриваемого случая является трёхкаскадная сеть с двумя свёрточными слоями в каждом каскаде и 128 картами признаков. В сравнении с двухкаскадной сетью

средняя эффективность снизилась более чем на 3%, но это обусловлено очень низкой эффективностью при малом числе признаков. Вместе с тем точность сети при 128 признаках составила 82,12%, что является лучшим результатом. Далее, в табл. 5 приведены результаты работы трёхкаскадной сети с тремя свёрточными слоями.

Таблица 5. Результаты работы трёхкаскадной нейронной сети с тремя свёрточными слоями в каждом каскаде

Table 5. The results of a three-stage neural network with three convolutional layers in each cascade

Число карт признаков	Первая итерация (%)	Вторая итерация (%)	Третья итерация (%)	Среднее значение (%)
8	50,79	44,70	46,10	47,20
16	64,59	64,55	64,43	64,52
24	71,66	71,11	72,19	71,65
32	75,79	74,32	75,30	75,14
40	76,82	77,73	75,81	76,79
48	79,30	80,33	78,25	79,29
64	77,98	80,37	79,70	79,35
128	81,49	78,62	80,13	80,08

Выводы

Наибольшее влияние на точность распознавания образов оказало увеличение числа карт признаков. Наращивание числа каскадов оказало менее заметный эффект, а увеличение числа свёрточных слоёв в каждом каскаде не всегда приводит к повышению точности работы нейронной сети.

На рис. 4 показана зависимость точности описанной выше нейронной сети на обучающей выборке (обозначена точками) и на проверочной выборке (обозначена линией) в зависимости от числа эпох обучения.

Сравнение результатов распознавания изображений представлено в табл. 6.

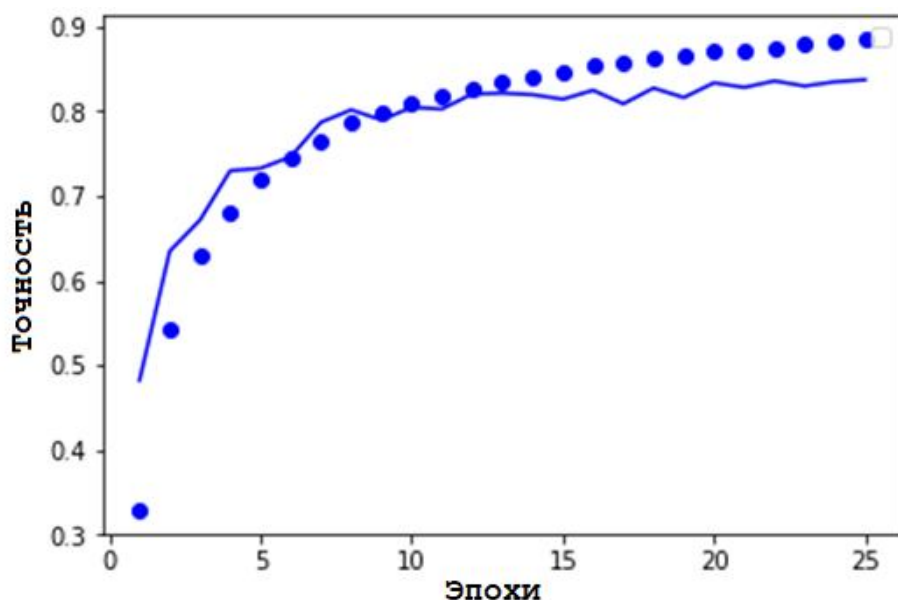


Рис. 4. Изменение точности трёхкаскадной нейронной сети с двумя свёрточными слоями в каждом каскаде в зависимости от числа обучающих эпох

Fig. 4. Changing the accuracy of a three-stage neural network with two convolutional layers in each stage depending on the number of training eras

Таблица 6. Сравнение результатов распознавания изображений

Table 6. Comparison of image recognition results

Изображение	3 каскада 2 слоя 8 признаков	3 каскада 2 слоя 128 признаков	3 каскада 3 слоя 8 признаков	3 каскада 3 слоя 128 признаков	2 каскада 2 слоя 8 признаков	2 каскада 2 слоя 128 признаков	2 каскада 3 слоя 8 признаков	2 каскада 3 слоя 128 признаков
	X	✓	X	X	✓	X	X	X
	✓	✓	X	✓	✓	✓	✓	✓
	✓	✓	X	✓	✓	✓	✓	✓
	✓	✓	X	✓	X	✓	X	X
	X	X	X	X	X	X	X	✓
	X	X	X	X	X	X	X	X

В соответствии с представленными данными лучший результат показывает трёхкаскадная сеть с двумя свёрточными слоями, верно распознав 4 изображения из 6 представленных. Таким образом подтверждено, что в рассматриваемом случае наиболее эффективной архитектурой нейронной сети оказалась трёхкаскадная сеть с двумя свёрточными слоями в каждом каскаде и 128 картами признаков.

Сравним точность данной модели с известными архитектурами свёрточных сетей, побеждавшими в ImageNet Large Scale Visual Classification Challenge (ILSVRC) с 2012 года. Оценка сетей производится по top-5 ошибке, это

означает, что ответ считается верным, если объект на изображении соответствует одному из 5 наиболее вероятных классов, выбранных сетью. Кроме того приводится ошибка ансамбля сетей, то есть группы отдельно обученных сетей, где ответом ансамбля считается среднее арифметическое ответов каждой отдельной сети. Так top-5 ошибки сети AlexNet для ансамбля в 2012 году с 15,40% уменьшился до 3,10% сети Inception-ResNet для ансамбля в 2016 году. Для одиночных моделей представленных нейронных сетей этот интервал составляет от 17% до 4,49%. Модель, полученную в эксперименте, будем сравнивать по top-1 ошибке, то есть за

ответ нейронной сети будем считать один класс с наибольшей вероятностью. Top-1 ошибки для полученной сети – значение, обратное точности. Таким об-

разом, $100\% - 82,12\% = 17,88\%$. В табл. 7 представлены результаты top-1 и top-5 ошибки для одной модели популярных архитектур нейронных сетей.

Таблица 7. Результаты top-1 и top-5 ошибки для одной модели популярных архитектур нейронных сетей

Table 7. Top-1 and top-5 errors results for one model of popular architectural neural networks

Нейронная сеть	Top-1	Top-5
AlexNet	39,00	17,00
ZF Net	37,50	16,00
VGG Net	25,60	8,10
GoogLeNet	29,00	9,20
Inception-v3	21,20	5,60
Inception-v4	20,00	5,00
Inception-ResNet-v2	19,90	4,90
ResNet-151	19,38	4,49

Видим, что полученная архитектура сети имеет лучший показатель top-1 ошибки среди представленных. Даль-

нейшее проведение эксперимента заключается в синтезе оптимальной архитектуры ансамбля таких сетей.

Список литературы

1. Гольцова И. А., Гуламов А. А. Информационное обеспечение участка железной дороги // Известия Юго-Западного государственного университета. Серия: Управление, вычислительная техника, информатика. Медицинское приборостроение. 2017. Т. 7, № 2(23). С. 6–11. https://swsu.ru/izvestiya/seriesivt/archiv/2_2017.pdf
2. Маклаков Е. С., Гуламов А.А. Узел сбора информации диспетчерского центра // Известия Юго-Западного государственного университета. 2018. Т. 22, № 6(81). С. 136-142. <https://doi.org/10.21869/2223-1560-2018-22-6-136-142>.
3. Маклаков Е. С., Гуламов А.А. Оптимизация «последних миль» до удаленных узлов доступа путем применения технологии LCAS // Моделирование, оптимизация и информационные технологии: научный журнал. 2019. Т. 7. № 3. <http://moit.vivt.ru/>. <https://doi.org/10.26102/2310-6018/2019.26.3.039>.
4. Видеосистема обнаружения морских судов по триангуляционным рашёткам / Ш. С. Фахми, Е. В. Костикова, М. С. Крюкова, С. А. Селиверстов // Морские интеллектуальные технологии. 2018. № 1-3 (41). С.143-155. http://morintex.ru/wp-content/files_mf/1536237135MITVOL41No3PART12018compressed.pdf
5. Фахми Ш. С., Шаталова Н. В., Крюкова М. С. Выделение контуров морских объектов на основе пирамидально-рекурсивного метода представления изображений //

Морские интеллектуальные технологии. 2019. № 2-2 (44). С.129-136. http://morintex.ru/wp-content/files_mf/1560970718MITVOL44No2PART12019.pdf

6. Выделение контуров изображений морских судов / Ш. С. Фахми, Н. В. Шаталова, С. А. Селиверстов, Е. С. Калинина, А. В. Иванов // Морские интеллектуальные технологии. 2019. № 3-3 (45). С.132-142. http://morintex.ru/wp-content/files_mf/1568625233MITVOL45No3PART32019_compressed1.pdf

7. Erhu Zhang, Kelu Wang, Guangfeng Lin Classification of marine vessels with multi-feature structure fusion // *Applied Science*. 2019. № 9(10). P. 2153. <https://doi.org/10.3390/app9102153>.

8. Tom Hope, Yehezkel S. Resheff, Itay Lieder Learning TensorFlow: A Guide to Building Deep Learning Systems *O'Reilly Media*; 1 edition, 2017. 242 p. <https://www.oreilly.com/catalog/errata.csp?isbn=0636920044116>

9. Андреас Мюллер. Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными. М.: Вильямс, 2017. 480 с. https://codernet.ru/books/python/vvedenie_v_mashinnoe_obuchenie_s_pomoshhyu_python/

10. Себастьян Рашка. Python и машинное обучение. М.: ДМК-Пресс, 2017. 418 с.

11. Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, Google Brain Tensor flow: A system for large-scale machine learning // Operating Systems Design and Implementation: Proc. 12th Symposium, Savannah, GA, USA, 2016. pp. 265-283. <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf>

12. Антонио Джулли, Суджит Пал. Библиотека Keras – инструмент глубокого обучения. Реализация нейронных сетей с помощью библиотек Theano и Tensor Flow. М.: ДМК-Пресс, 2017. 296 с.

13. Франсуа Шолле. Глубокое обучение на Python. СПб.: Питер, 2018. 400 с.

14. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems *O'Reilly Media*, 2017. 574 p. https://www.academia.edu/37010160/Hands-On_Machine_Learning_with_Scikit-Learn_and_TensorFlow.

15. Ian Goodfellow, Deep Learning (Adaptive Computation and Machine Learning series) *The MIT Press*, 2016. 800 p. https://www.academia.edu/38223830/Adaptive_Computation_and_Machine_Learning_series-Deep_learning-The_MIT_Press_2016_.pdf

16. Tariq Rashid, Make Your Own Neural Network *CreateSpace Independent Publishing Platform*, 2016. 222 p.

17. Schmidhuber J. Deep learning in neural networks: An overview // *Neural Networks*. 2015. Vol. (61). P. 85–117. <http://www2.econ.iastate.edu/tesfatsi/DeepLearningInNeuralNetworksOverview.JSchmidhuber2015.pdf>

18. Josh Patterson, Adam Gibson Deep Learning: A Practitioner's Approach *O'Reilly Media, Inc.* 2017. 532 p. https://www.academia.edu/37119738/Deep_Learning_A_Practitioners_Approach
19. Саймон Хайкин. Нейронные сети. М.: Вильямс, 2018. 1104 с.
20. Michael Taylor The Math of Neural Networks *Amazon Digital Services LLC - Kdp Print Us*, 2017. 168 p. https://cours.etsmtl.ca/sys843/REFS/Books/ebook_Haykin09.pdf

References

1. Gol'tsova I. A., Gulamov A. A. Informationsionnoe obespechenie uchastka zheleznoi dorogi [Information support of the railway section]. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta. Seriya: Upravlenie, vychislitel'naya tekhnika, informatika. Meditsinskoe priborostroenie = Proceedings of the Southwest State University. Series: Control, Computing Engineering, Information Science. Medical Instruments Engineering*, 2017, vol. 7, no. 2(23), pp. 6–11 (In Russ.).
2. Maklakov Ye. S., Gulamov A.A. Uzel sbora informatsii dispetcherskogo tsentra [The Collection of Information Control Center]. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University*, 2018, vol. 22, no. 6(81), pp. 136-142 (In Russ.). <https://doi.org/10.21869/2223-1560-2018-22-6-136-142> (In Russ.).
3. Maklakov Ye. S., Gulamov A.A. Optimizatsiya «poslednikh mil'» do udalennykh uzlov dostupa putem primeneniya tekhnologii LCAS [Optimization of "last miles" to remote access nodes by application of LCAS technology]. *Modelirovaniye, optimizatsiya i informatsionnyye tekhnologii. Nauchnyy zhurnal = Modeling, optimization and information technology. Scientific journal*, 2019, vol. 7, no. 3. <https://doi.org/10.26102/2310-6018/2019.26.3.039> (In Russ.).
4. Fakhmi Sh. S., Kostikova E. V., Kryukova M. S., Seliverstov S. A. Videosistema obnaruzheniya morskikh sudov po triangulyatsionnym rashetkam [Video system for the detection of ships by triangulation grids]. *Morskiye intellektual'nyye tekhnologii = Marine Intelligent Technology*, 2018. no. 1-3 (41), pp.143-155. http://morintex.ru/wp-content/files_mf/1536237135MITVOL41No3PART12018compressed.pdf (In Russ.).
5. Fakhmi Sh. S., Shatalova N. V., Kryukova M. S. Vyделение konturov morskikh ob"ektov na osnove piramidal'no-rekursivnogo metoda predstavleniya izobrazhenii [Isolation of the contours of marine objects based on the pyramidal-recursive image representation method]. *Morskiye intellektual'nyye tekhnologii = Marine Intelligent Technology Publ.*, 2019, no. 2-2 (44), pp.129-136. http://morintex.ru/wp-content/files_mf/1560970718MITVOL44No2PART12019.pdf (In Russ.).
6. Fakhmi Sh. S., Shatalova N. V., Seliverstov S. A., Kalinina E. S., Ivanov A. V. Vyделение konturov izobrazhenii morskikh sudov [Highlighting the contours of images of ships]. *Morskiye intellektual'nyye tekhnologii = Marine Intelligent Technology*. 2019, no. 3-3 (45), pp.132-142.

7. Erhu Zhang, Kelu Wang, Guangfeng Lin Classification of marine vessels with multi-feature structure fusion. *Applied Science*, 2019, no. 9(10), 2153 p. <https://doi.org/10.3390/app9102153>.

8. Tom Hope, Yehezkel S. Resheff, Itay Lieder Learning TensorFlow: A Guide to Building Deep Learning Systems *O'Reilly Media*; 1 edition, 2017, 242 pp. <https://www.oreilly.com/catalog/errata.csp?isbn=0636920044116>.

9. Andreas Myuller. *Vvedenie v mashinnoe obuchenie s pomoshch'yu Python* [Introduction to machine learning with Python. Guide for data professionals]. Moscow, Vil'yams Publ., 2017, 480 p. https://codernet.ru/books/python/vvedenie_v_mashinnoe_obuchenie_s_pomoshhyu_python/ (In Russ.).

10. Sebast'yan Rashka. *Python i mashinnoe obuchenie* [Python and machine learning]. Moscow, *DMK-Press Publ.*, 2017. 418 p. (In Russ.).

11. Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, Google Brain Tensor flow: A system for large-scale machine learning. Operating Systems Design and Implementation: Proc. 12th Symposium, Savannah, GA, USA, 2016, pp. 265-283. <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf>.

12. Antonio Dzhulli, Sudzhit Pal. *Biblioteka Keras – instrument glubokogo obucheniya. Realizatsiya neironnykh setei s pomoshch'yu bibliotek Theano i Tensor Flow* [Keras Library is a deep learning tool. Implementing neural networks using Theano and Tensor Flow libraries.]. Moscow, *DMK-Press Publ.*, 2017, 296 p. (In Russ.).

13. Fransua Sholle. *Glubokoe obuchenie na Python* [Python Deep Learning]. Saint-Petersburg, Piter Publ., 2018. 400 p. (In Russ.).

14. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems *O'Reilly Media*, 2017. 574 p. https://www.academia.edu/37010160/Hands-On_Machine_Learning_with_Scikit-Learn_and_TensorFlow.

15. Ian Goodfellow, Deep Learning (Adaptive Computation and Machine Learning series) *The MIT Press*, 2016. 800 p. https://www.academia.edu/38223830/Adaptive_Computation_and_Machine_Learning_series-Deep_learning-The_MIT_Press_2016.pdf

16. Tariq Rashid, Make Your Own Neural Network *CreateSpace Independent Publishing Platform*, 2016, 222 p.

17. Schmidhuber J. Deep learning in neural networks: An overview *Neural Networks*. 2015, vol. (61), pp. 85–117. <http://www2.econ.iastate.edu/tesfatsi/DeepLearningInNeural-NetworksOverview.JSchmidhuber2015.pdf>.

18. Josh Patterson, Adam Gibson *Deep Learning: A Practitioner's Approach* *O'Reilly Media, Inc.* 2017, 532 p. https://www.academia.edu/37119738/Deep_Learning_A_Practitioners_Approach.
 19. Saimon Khaikin. *Neironnye seti* [Neural networks]. Moscow, Vil'yams Publ., 2018, 1104 p. (In Russ.).
 20. Michael Taylor *The Math of Neural Networks* *Amazon Digital Services LLC - Kdp Print Us*, 2017. 168 p. https://cours.etsmtl.ca/sys843/REFS/Books/ebook_Haykin09.pdf
-

Информация об авторах / Information about the Authors

Конарев Дмитрий Игоревич, аспирант,
ФГБОУ ВО «Юго-Западный
государственный университет»,
г. Курск, Российская Федерация,
e-mail: dmitrii.konarev@gmail.com

Dmitrii I. Konarev, Post-Graduate Student,
Southwest State University,
Kursk, Russian Federation,
e-mail: dmitrii.konarev@gmail.com

Гуламов Алишер Абдумаликович,
доктор физико-математических наук,
доцент, профессор кафедры КПиСС,
ФГБОУ ВО «Юго-Западный
государственный университет»,
г. Курск, Российская Федерация,
e-mail: profgulamov@mail.ru

Alisher A. Gulamov, Dr. of Sci. (Engineering),
Associate Professor, Southwest State
University, Kursk, Russian Federation,
e-mail: profgulamov@mail.ru