

**СРАВНИТЕЛЬНЫЙ АНАЛИЗ АРХИТЕКТУР СВЕРТОЧНЫХ
НЕЙРОННЫХ СЕТЕЙ
COMPARATIVE ANALYSIS OF CONVOLUTIONAL NEURAL
NETWORK ARCHITECTURES**

Шестаков Д.В., студент

Евдокимова С.А., к.т.н., доцент

ФГБОУ ВО «Воронежский государственный лесотехнический университет
имени Г.Ф. Морозова»

г. Воронеж, Россия

dmitrys4estakov@yandex.ru

Shestakov D.V., student

Evdokimova S.A., CSc (Engineering), Associate Professor

FSBEI HE «Voronezh State University of Forestry and Technologies named
after G.F. Morozov»

Voronezh, Russian Federation

Аннотация: В работе проводится сравнение наиболее популярных архитектур сверточных нейронных сетей, их преимущества и недостатки, области применения. Рассматриваются такие нейронные сети, как ALEXNET, ZFNET, VGG, GOOGLNET, RESNET.

Summary: The paper compares the most popular architectures of convolutional neural networks, their advantages and disadvantages, and areas of application. Neural networks such as ALEX NET, ZF NET, VG, GOOGLNET, RESET are analyzed.

Ключевые слова: Свёрточная нейронная сеть, машинное обучение, свертка, фильтр, подвыборка, функция активации.

Keywords: Convolutional neural network, machine learning, filter, subsample, activation function.

Свёрточные нейронные сети (CNN) стали большим достижением в области компьютерного зрения, с развитием машинного обучения CNN продемон-

стрировали впечатляющую эффективность в решении широкого спектра задач [1].

CNN была предложена Яном Лекуном в 1988 году [2]. Стоит сказать, что более успешно сверточные сети применяются как раз в работе с изображениями и видео. CNN бывают одномерными, такие сети работают с временными рядами, аудио и текстом, а также применяются для обработки естественного языка. Но сама суть сверточных нейронных сетей и их основное отличие заключается в работе с двумерными и трехмерными данными. На эту роль подходят как раз изображения и видео (третья мера – время).

Модель нейрона может быть представлена следующим образом (рисунок 1) [3]: имеется I – множество нейронов сети; $V = \{(i, j) | i, j \in N\}$ – множество связей между входным i -ым и выходным j -ым нейроном; $w: V \rightarrow R, w(i, j)$ или $w_{i,j}$ – веса связей между нейронами i и j . К j -ому нейрону функция активации доставляет выходы $x_1, x_2, \dots, x_n$ нейронов, имеющих связь с j -ым нейроном, и присоединяет к ним веса $w_{i,j}$, формируя вход нейрона net_j , определяемый по формуле:

$$net_j = \sum_{i=1}^n w_{i,j} \cdot x_i + w_0, \quad (1)$$

где w_0 – смещение.

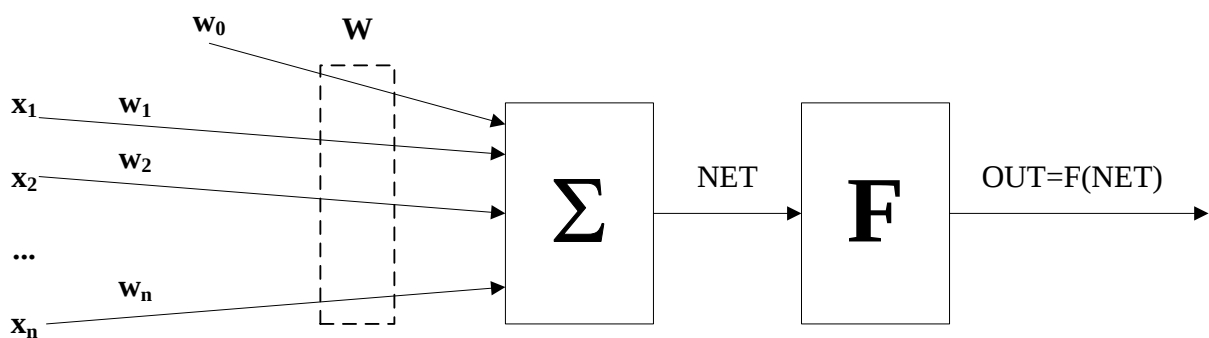


Рисунок 1 – Модель нейрона

На поступивший сигнал реакция каждого нейрона определяется с помощью функции активации:

$$a_j = F(net_j, a_{j-1}, b_j), \quad (2)$$

где a_j – состояние активации j -го нейрона;

b_j – пороговое значение j -го нейрона, определяемое как максимальное значение градиента функции активации.

Структура свёрточной нейронной сети состоит из следующих слоев [4]:

1. Входной слой (Input layer), основная задача которого заключается в получении изображения в виде массива чисел. Любые рисунки или видеокадры представляют собой упорядоченный набор пикселей, каждый пиксель содержит набор значений от 0 до 255, описывающих цвет (RGB) и его интенсивность.

2. Свёрточный слой (Convolutional layer) позволяет выделить важные детали изображения, а всю лишнюю информацию (например, фон) убрать. Фильтры представляют собой матрицы, которые скользят по изображению и создают новую матрицу с помощью различных функций (рисунок 2). Элементы ядра называются весами свёрточного слоя, они настраиваются в процессе обучения согласно заданному оптимизатору. В большинстве алгоритмов дополнительно к свертке используется операция суммирования для того, чтобы при нулевых входных данных свертки получить результат, отличный от нуля.

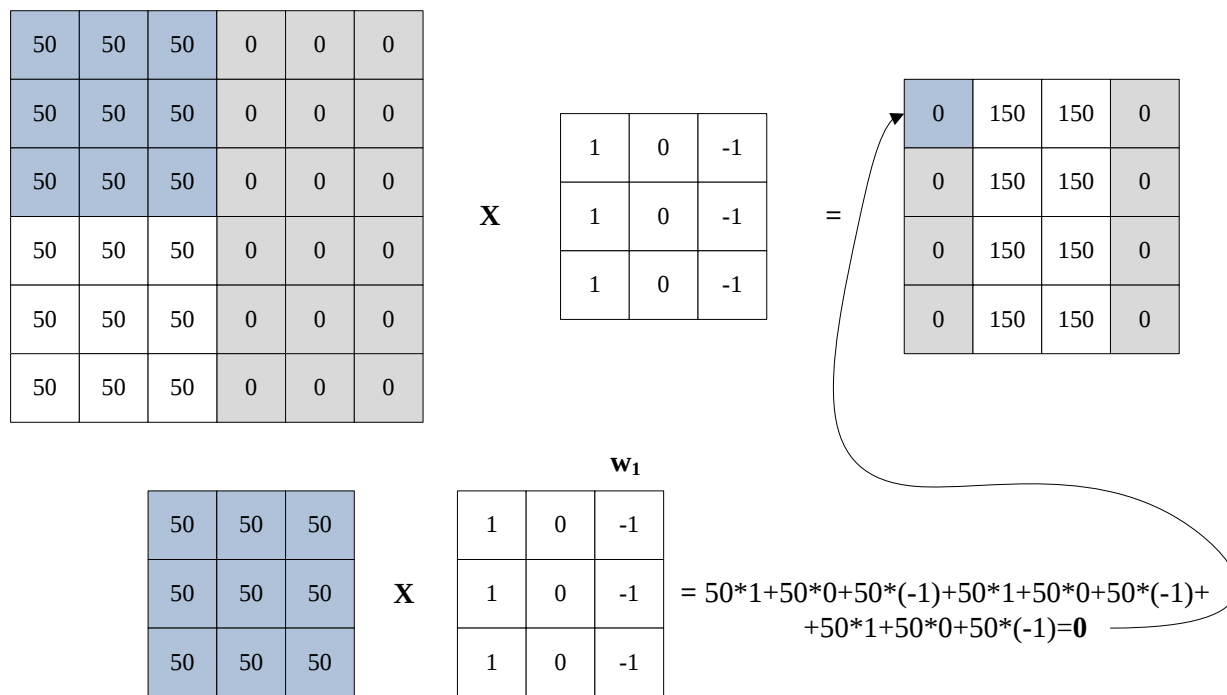


Рисунок 2 – Слой свертки

3. Слой активации (Activation layer) располагается между любыми двумя слоями и использует простые функции активации, которые необходимы для добавления нелинейности в модель, что улучшает и ускоряет работу сети с минимальными затратами.

4. Пулинговый слой (Pooling layer) уменьшает размерность карт признаков, сохраняя при этом важные свойства и снижая вычислительные затраты. Это помогает нейросети эффективно обрабатывать большие изображения. Существуют несколько способов выполнения пулинга: по максимальному (рису-

нок 3) или среднему значению в матрице заданного размера ядра (рисунок 4), взвешенному среднему значению на основании расстояния относительно центрального пикселя и т.д. [5]. Используя комбинацию слоев свертки и слоев пулинга получается основной структурный элемент свёрточной нейронной сети.

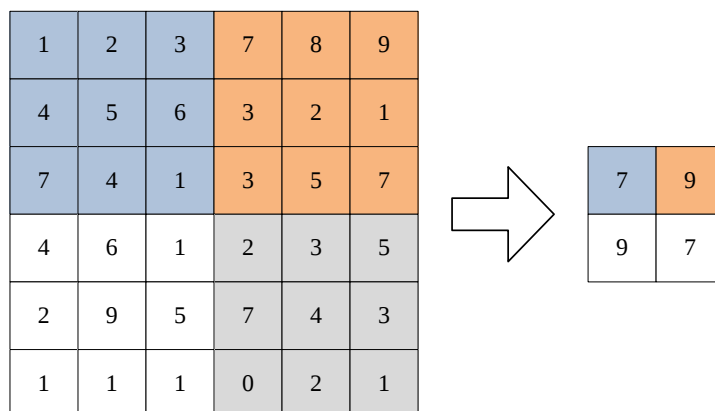


Рисунок 3 – Пулинг по максимальному значению

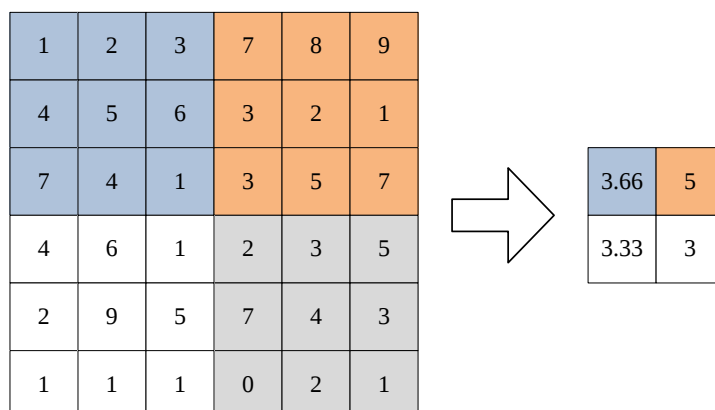


Рисунок 4 – Пулинг по среднему значению

5. Выпрямляющий слой (Flattening layer) – этот слой просто преобразует карты признаков в одномерный массив для дальнейшей обработки в следующем слое.

6. Полносвязные слои (Fully Connected layers) соединяют все нейроны текущего слоя с нейронами предыдущего, что позволяет комбинировать все признаки для окончательной классификации. В результате двумерная карта признаков преобразуется в одномерный вектор и получается финальный результат.

7. Выходной слой (Output layer). На этом этапе вычисляется вероятность принадлежности изображения к определенному классу.

Обработка изображений с помощью CNN имеет следующие особенности [4]:

1. Изображения представляют собой объекты с большим количеством признаков;

2. Изображение в формате RGB размера 640×480 будет иметь ~1млн признаков, потому что признаком является не только каждый пиксель, но и его интенсивность;

3. В отличие от полносвязной сети, каждый нейрон свёрточного слоя принимает на вход только часть исходного изображения;

4. Операция свёртки требовательна к вычислительным ресурсам;

5. Для свертки RGB изображения размера 256×256 фильтром 5×5 необходимо выполнить ~5млн операций умножения;

6. На практике операцию свертки сводят к перемножению матриц;

7. В отличие от полносвязных, свёрточные сети содержат меньше параметров, поэтому они менее склонны к переобучению.

Для того чтобы модель работала, необходимо ее обучить. Это значит, что для того, чтобы она понимала, что изображено на картинке, ей сначала нужно эти картинки «показать». Для обучения модели выбирается функция ошибки и оптимизатор. Функция ошибки определяет разницу между предсказаниями модели и истинными значениями меток. Оптимизатор вычисляет оптимальные веса модели таким образом, чтобы минимизировать ошибки.

При сравнении архитектур свёрточных нейронных сетей будем рассматривать их следующие характеристики [3, 4]:

1. Глубина сети: количество слоев в сети, включая свёрточные, пулинговые и полносвязные слои. Глубокие сети могут лучше улавливать сложные паттерны, но могут быть более подвержены проблемам переобучения и требуют больше вычислительных ресурсов.

2. Типы слоев: типы используемых слоев, такие как свёрточные, пулинговые, полносвязные, нормализационные слои и слои с регуляризацией. Различные комбинации этих слоев могут влиять на способность сети к обобщению и устойчивость к шуму.

3. Размер фильтров: размеры фильтров (ядер) в свёрточных слоях, такие как 3×3 , 5×5 или 11×11 . Более крупные фильтры могут улавливать более общие признаки, но требуют больше вычислительных ресурсов.

4. Функции активации: виды используемых функций активации, таких как ReLU, Leaky ReLU, Sigmoid или Tanh. Выбор функции активации может влиять на скорость сходимости и способность сети к обучению.

5. Регуляризация: методы регуляризации, такие как Dropout, Batch Normalization или L2 регуляризация. Регуляризация помогает предотвратить переобучение и улучшить обобщающую способность модели.

6. Архитектурные инновации: наличие специализированных архитектурных компонентов, таких как Residual Blocks (в ResNet), Inception модули (в GoogLeNet) или Dense Blocks (в DenseNet).

7. Количество параметров: общее количество обучаемых параметров в сети. Более сложные сети с большим количеством параметров могут лучше улавливать сложные паттерны, но требуют больше данных и вычислительных ресурсов для обучения.

8. Скорость и эффективность: время выполнения одного прохода и обратного прохода, а также использование памяти. Эти факторы важны для оценки применимости сети в реальных условиях.

Для удобства будем рассматривать нейронные сети по мере их появления и по возрастанию их мощностей.

AlexNet – это сверточная нейронная сеть, разработанная А. Крижевским, И. Суцкевером и Д. Хинтоном в 2012 г. Она была представлена на конкурсе ImageNet 2012 и одержала в нем победу, благодаря использованию глубоких архитектур и графических процессоров (GPU) [6].

Архитектура AlexNet состоит из восьми слоев, пять из которых являются сверточными слоями, а три – полносвязными. AlexNet был обучен на большом наборе данных ImageNet, который содержит миллионы изображений с тысячами классов объектов. Благодаря этому сеть смогла достичь высокой точности в классификации изображений. Ошибка AlexNet составляет 15,3% [6].

В AlexNet используется функция активация Relu, определяемая как:

$$F(x) = \begin{cases} x, & \text{если } x \geq 0; \\ 0, & \text{иначе.} \end{cases}, \quad (3)$$

применяется после каждого свёрточного и полносвязного слоя. Дропаут (Dropout) используется перед первым и вторым полносвязными слоями.

AlexNet проходит 90 эпох обучения, для оптимизации параметров модели используется метод SGD (Stochastic Gradient Descent) с импульсом, что позволяет быстрее скорректировать направление очередного шага.

Сегодня AlexNet уже считается устаревшей моделью, однако она заложила основу для дальнейшего развития нейронных сетей, таких как VGG, ResNet и другие современные архитектуры.

ZFNet представляет собой модификацию сети AlexNet, разработанная с целью улучшения ее архитектуры путем более глубокого анализа ошибок, возникающих на этапе обучения. В 2013 году ZFNet победила на соревновании ILSVRC с ошибкой 11,2% [7]. Этого удалось достичь благодаря точной

настройке гиперпараметров (размер фильтров и их количество, скорость обучения, размер пакетов и т. д.). В ZFNet в первом свёрточном слое размер фильтра равен 7×7 , шаг – 2, а количество фильтров в чистых свёрточных слоях сети (3, 4, 5).

Разработчики ZFNet М. Цилер и Р. Фергюс предложили систему визуализации активаций промежуточных слоев сети для понимания того, какие именно особенности изображений наиболее важны для классификации. Это помогло им выявить ошибки в архитектуре и оптимизировать ее параметры.

Идеи, заложенные в сетях ZFNet, были использованы в VGG и ResNet.

VGG (VisualGeometryGroup) – это семейство свёрточных нейронных сетей, разработанных исследовательской группой VisualGeometryGroup Оксфордского университета в 2014 году [8].

Основной особенностью архитектуры VGG является использование небольших фильтров 3×3 , которые применяются последовательно через несколько слоев, что позволяет более точно выделять детали изображения. VGG использует пулинг по максимальному значению с окном 2×2 и шагом 2. Так же, как и AlexNet, в VGG применяется функция активации Relu, что позволяет сети быстрее обучаться. После нескольких свёрточных слоев идут два полносвязных слоя с 4096 нейронами каждый, а третий слой – 1000 нейронов. Количество ошибок было сокращено до 7,3%.

В настоящее время существует несколько версий сети VGG, различающихся глубиной и количеством параметров: VGG11 (8 сверток, 3 полносвязных слоя); VGG13 (добавлены дополнительные свёрточные слои); VGG16 (13 сверток, 3 полносвязных слоя); VGG19 (16 сверток). Самой популярной версией является VGG16.

Недостатками VGG является большое количество параметров модели, что требует значительных вычислительных ресурсов, и медленная скорость работы.

GoogLeNet (CNN от компании Google) была разработана в 2014 году и стала победителем конкурса ImageNet с ошибкой 6,7%. GoogLeNet использует модуль Inception, который объединяет несколько типов сверток и пулинга внутри одного блока. Это позволило значительно увеличить глубину модели, сохраняя при этом относительно небольшое количество параметров.

Структура модуля Inception включает параллельные потоки обработки, где выполняются различные операции над входящими данными, включая свертки разного размера и пулинги. Результаты всех потоков объединяются по размеру канала, чтобы сформировать выходные данные, передаваемые следу-

ющему блоку [9]. Затем исходные блоки накладываются друг на друга, по одному слою за раз. Таким образом, вводится разреженный тип и моделируется оптимальная топология нейронной сети. Для уменьшения общего количества параметров, в начальный блок добавляется свертка 1×1 , что эффективно уменьшает размер объекта и позволяет избежать увеличения количества параметров.

Архитектура сети GoogleNet состоит из 22 слоев с параметрами, но общее количество слоев, используемых для построения сети, составляет 100.

В качестве недостатков GoogleNet можно отметить сложность понимания и интерпретации многослойных модулей Inception, затухание градиентов по мере прохождения через слои, большое количество гиперпараметров и высокие требования к вычислительным ресурсам. Эти ошибки разработчики постарались исправить в следующих версиях Inception v2, v3 и v4.

ResNet (Residual Network, в переводе с англ. – остаточная сеть) разработана исследователями из Microsoft Research Asia в 2015 году, сеть использует остаточные связи, позволяющие строить чрезвычайно глубокие сети, избегая проблем, связанных с затуханием градиента и переобучением [10]. Эти свойства делают ResNet особенно эффективной для решения задач классификации изображений, обнаружения объектов и сегментации.

Остаточный блок работает следующим образом:

$$y = F(x) + x; , \quad (4)$$

где добавление исходного сигнала x к результату преобразования $F(x)$ позволяет сети учиться представлять небольшие изменения сигнала, а не полное отображение входа в выход. Остаточный блок включает в себя несколько свёрточных слоев, к которым добавляется прямое соединение, проходящее через весь блок. В остаточных блоках используются свёрточные слои с фильтрами 1×1 .

Благодаря остаточным связям ResNet может иметь сотни слоев, благодаря остаточным связям. В более глубоких сетях ResNet остаточный блок состоит из трех сверток: сначала для сокращения размерности 1×1 , затем для свертки 3×3 и вновь 1×1 для увеличения размерности.

ResNet обеспечивает высокую точность результатов, например, ResNet-152 достигает ошибки 3.57% [11].

ResNet нашла широкое применение в различных задачах компьютерного зрения: классификации изображений, детекции объектов, сегментации, анализе видео, распознавании действий и отслеживании объектов.

На рисунке 4 показана диаграмма сравнения точности работы CNN при решении задач распознавания изображений.

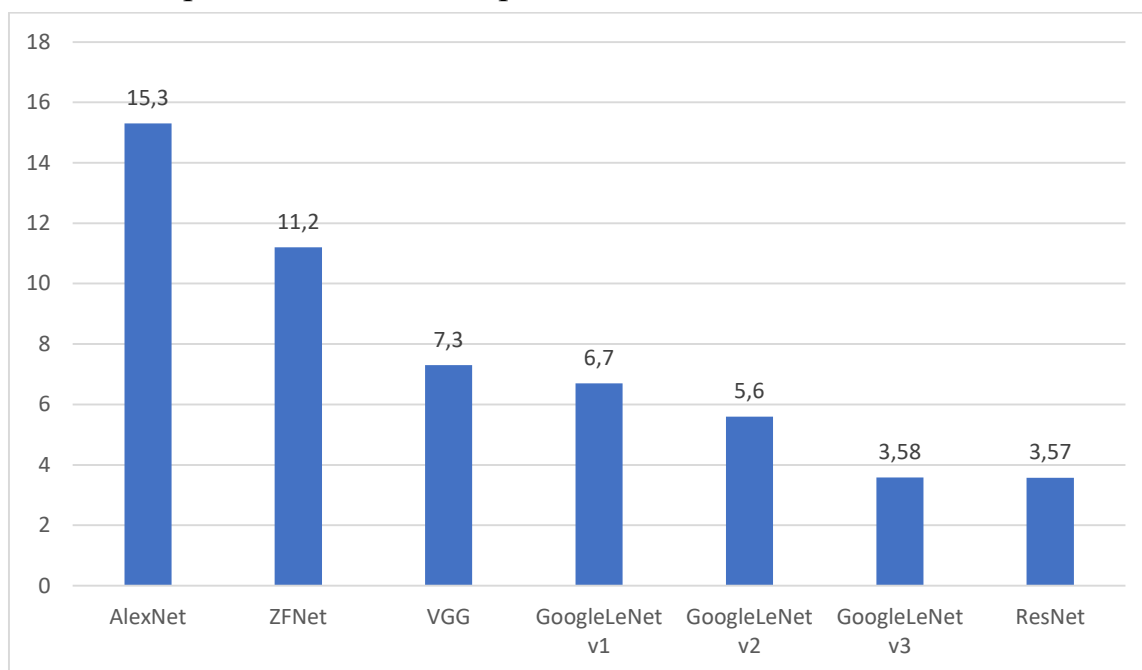


Рисунок 1 – Сравнение ошибок (%) CNN в задачах распознавания изображений

В результате проведенного сравнения лучший результат показывает модель с архитектурой ResNet – 3,57% ошибки. Однако, изучив параметры этой сети, можно сказать, что модель требует больше вычислительных ресурсов и подвержена проблемам переобучения, поэтому стоит учитывать особенности области применения.

Список литературы

1. Сикорский, О.С. Обзор сверточных нейронных сетей для задачи классификации изображений / О.С. Сикорский // Новые информационные технологии в автоматизированных системах. – 2017. – № 20. – С. 37-42.
2. Маршалко, Д.А. Архитектура сверточных нейронных сетей / Д.А. Маршалко, О.В. Кубанских // Ученые записки Брянского государственного университета. – 2019. - № 4. – С. 10-13.
3. Сверточная нейронная сеть, часть 1: структура, топология, функции активации и обучающее множество. – URL: <https://habr.com/ru/articles/348000/> (дата обращения: 12.03.2025).
4. Саймон, Х. Нейронные сети: полный курс / Х. Саймон. – М. : Издательский дом «Вильямс», 2019. – 1104 с.

5. Макаров, М.А. Искусственный интеллект в промышленной информатике : учебное пособие / М.А. Макаров, И.Ю. Зайцев. – М. : МИРЭА – РТУ, 2024. – 118 с.
6. Zhou, L. Design and Implementation of AlexNet Flower Classification System from the Perspective of Deep Learning / L. Zhou // International Journal of Computer Science and Information Technology. – 2024. – Vol. 3(3). – Pp. 292-299.
7. Эволюция архитектур нейросетей в компьютерном зрении: классификация изображений. – URL: <https://habr.com/ru/companies/slsoft/articles/855602/> (дата обращения: 12.03.2025).
8. Face Images Classification using VGG-CNN / I.N.G.A. Astawa, M.L. Radhitya, I.W.R. Ardana, F.A. Dwiyanto // Knowledge Engineering and Data Science. – 2021. – Vol. 4(1). – Pp. 49-54.
9. Dong, H. Investigations on Convolutional Neural Network Based on Image Recognition / H. Dong // Highlights in Science, Engineering and Technology. – 2025. – Vol. 124. – Pp. 208-215.
10. Чжу, И. Исследование остаточной нейронной сети RESNET / И. Чжу, Т.П. Новикова, С.А. Евдокимова // Моделирование информационных систем и технологий : материалы Международной научно-практической конференции, Воронеж, 27 октября 2022 г. – Воронеж, 2022. – С. 343-349.
11. Deep Residual Learning for Image Recognition / K. He [et al.] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2016. – Pp. 770-778.

References

1. Sikorsky, O.S. Overview of convolutional neural networks for image classification / O.S. Sikorsky // New information technologies in automated systems. – 2017. – No. 20. – Pp. 37-42.
2. Marshalko, D.A. Architecture of convolutional neural networks / D.A. Marshalko, O.V. Kubanskikh // Scientific Notes of Bryansk State University. – 2019. – No. 4. – Pp. 10-13.
3. Convolutional neural network, part 1: structure, topology, activation functions and training set. – URL: <https://habr.com/ru/articles/348000/> (date of access: 03/12/2025).
4. Simon, H. Neural networks: a complete course / H. Simon. – M. : Williams Publishing House, 2019. – 1104 p.

5. Makarov, M.A. Artificial intelligence in industrial computer science: a textbook / M.A. Makarov, I.Y. Zaitsev. – M. : MIREA – RTU, 2024. – 118 p .
6. Zhou, L. Design and Implementation of AlexNet Flower Classification System from the Perspective of Deep Learning / L. Zhou // International Journal of Computer Science and Information Technology. – 2024. – Vol. 3(3). – Pp. 292-299.
7. Evolution of neural network architectures in computer vision: classification of images. – URL: <https://habr.com/ru/companies/slsoft/articles/855602/> / (date of access: 12/03/2025).
8. Face Images Classification using VGG-CNN / I.N.G.A. Astawa, M.L. Radhitya, I.W.R. Ardana, F.A. Dwiyanto // Knowledge Engineering and Data Science. – 2021. – Vol. 4(1). – Pp. 49-54.
9. Dong, H. Investigations on Convolutional Neural Network Based on Image Recognition / H. Dong // Highlights in Science, Engineering and Technology. – 2025. – Vol. 124. – Pp. 208-215.
10. Zhu, I. Investigation of the residual neural network RESNET / I. Zhu, T.P. Novikova, S.A. Evdokimova // Modeling of information systems and technologies: proceedings of the International Scientific and Practical Conference, Voronezh, October 27, 2022. – Voronezh, 2022. – Pp. 343-349.
11. Deep Residual Learning for Image Recognition / K. He [et al.] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2016. – Pp. 770-778.