

Сверточная нейронная сеть для сопоставления изображений товаров

В. А. Зенков¹, В. С. Панищев¹, А. Л. Желанов², Д. А. Волков³

¹Юго-Западный государственный университет, г. Курск

²Курская академия государственной и муниципальной службы, г. Курск

³Российский государственный университет нефти и газа (национальный исследовательский университет) имени И.М. Губкина, г. Москва

Аннотация: Статья посвящена применению сверточных нейронных сетей для сопоставления изображений товаров. Описана актуальность разработки систем сопоставления изображений товаров. Проведен анализ методов сравнения изображений, отмечены преимущества и недостатки каждого метода. Выполнено сопоставление изображений товаров торговых сетей с использованием нейронной сети ResNet. Проведены эксперименты и тестирование нейронных сетей ResNet на наборах данных товаров торговых сетей для сопоставления изображений, исследована точность сопоставления изображений для разных архитектур нейронной сети ResNet. Сделаны выводы о возможности применения нейронной сети ResNet для сопоставления изображений товаров.

Ключевые слова: сверточная нейронная сеть, распознавание, изображение, анализ изображений, соответствие изображению, ResNet.

Введение

В последнее время отмечается рост популярности технологий для распознавания товаров среди крупных представителей бизнеса. Решения такого рода находят свое применение у производителей товаров народного потребления, торговых сетей, оптовиков, импортеров, розничных магазинов в формате у дома.

Среди причин использования технологий подобного рода следует выделить следующие:

- уменьшение временных затрат на аудит и контроль за соблюдением стандартов,
- повышение клиентского сервиса путем получения оперативной информации о товаре при помощи смартфона,
- снижение расходов на персонал посредством уменьшения ручного труда: поиск информации о товаре, применение «умных весов» [1].

Примерами бизнесов, применяющих данные технологии, могут служить ритейлеры «Магнит» и «Лента», которые активно внедряют технологии распознавания товаров на полках [1,2].

Актуальность предложенной темы обусловлена тем, что существующие системы в области распознавания изображений товаров не являются универсальными и зачастую разработаны для решения задач конкретного производства или бизнеса. Данный недостаток влечет за собой сложности в поддержке и тиражировании, а также требует создания различных систем для частных случаев применения.

Решаемая в работе задача заключается в исследовании возможности применения нейронных сетей архитектуры ResNet для сопоставления изображений товаров. Применение нейронных сетей данной архитектуры должно обеспечить точное и быстрое сопоставление изображений, упростить процесс проверки соответствия товаров и повысить эффективность работы пользователей в области сравнения товаров.

Обзор методов сравнения изображений

На основе обзора литературы были выделены основные методы сопоставления изображений. Такими методами являются: использование векторов признаков, использование перцепционных хэш-функций, сравнение гистограмм RGB. Далее приводится описание данных методов, а также выделяются их положительные качества и недостатки:

1. Сравнение гистограмм RGB.

Метод основан на создании и сравнении гистограмм RGB, отражающих распределение яркости по каналам для каждого изображения. Гистограмма в анализе изображений представляет собой графическое представление распределения пикселей изображения, которое описывает количество или частоту различных значений интенсивности. Важно также отметить, что гистограмма не содержит пространственной информации о

соответствующем изображении, то есть изображение не может быть восстановлено из гистограммы, и два разных изображения могут иметь одинаковые гистограммы. Таким образом, гистограммы могут быть идентичными для двух различных изображений, содержащих разные объекты, но обладающих одинаковой цветовой информацией. Схожие гистограммы указывают на визуальное сходство изображений [3, 4].

Преимущества:

- устойчивость к трансформациям (поворот изображения, изменения размера изображения), не сильно изменяющим гистограмму.

Недостатки:

- не учитывается расположение и структура объектов на изображении, что может привести к ложному сопоставлению из-за одинаковых цветовых распределений на различных изображениях.

2. Перцепционные хэш-функции.

Представляют собой алгоритмы для вычисления хэшей на основе отдельных признаков изображения [5]. Вычисленный на основе алгоритмов хэш является в некотором виде отпечатком изображения. При сравнении изображений значения хэшей сравниваются с использованием специальных функций (метрик), которые позволяют вычислить коэффициент различия или сходства изображений. Заключение о схожести изображений основывается на выбранном пороговом значении. Непосредственно процесс сравнения изображений состоит из следующих шагов: уменьшение размера изображения, перевод в градации серого, вычисление среднего значения пикселей, создание бинарного изображения и получение хэша.

Преимущества:

- малый размер хэша, скорость вычисления и поиска,
- устойчивость к изменениям размеров изображения.

Недостатки:

- неустойчивость к трансформациям, таким как поворот или обрезка изображения.

3. Использование векторов признаков.

Сравнение векторов признаков (от англ. features vectors comparison) — более сложный, но эффективный способ поиска похожих изображений. Вектор признаков — это упорядоченный набор числовых значений, каждое из которых соответствует определенному признаку объекта. Этот вектор представляет объект в многомерном пространстве признаков. Сравнение векторов признаков имеет ряд преимуществ перед другими методами сравнения изображений, поскольку векторы признаков могут быть устойчивыми к изменениям масштаба, освещения, повороту и другим типичным трансформациям изображений.

Для извлечения векторов признаков из изображений допускается использовать нейронную сеть любой архитектуры, требуется только наличие на выходе нейросети массива числовых значений, представляющего собой вектор признаков и использующегося для классификации изображений. Наибольшую эффективность в вопросе извлечения векторов признаков изображений показали свёрточные нейронные сети (от англ. Convolutional Neural Network, далее CNN) [6,7], так как CNN наиболее подходят для работы с изображениями, как со стороны производительности сети, так и в силу того, что каждый уровень иерархии в CNN способен извлекать и представлять все более сложные и абстрактные признаки, делая представление данных гораздо богаче и информативнее. В целом, CNN — это нейронные сети, в которых вместо общей операции умножения на матрицу, по крайней мере в одном слое, используется свертка [8-10].

Преимущества:

- устойчивость к изменениям масштаба, освещения, повороту и другим трансформациям.
-

Недостатки:

- большая вычислительная сложность,
- зависимость от качества и архитектуры используемой нейронной сети.

Сопоставление изображений с использованием нейронной сети ResNet

Для решения задачи сопоставления изображений товаров требуется решение, устойчивое к различным изменениям: цвета, освещения, угла съемки и масштаба. Сложные данные изображений товаров могут включать разнообразие упаковок, этикеток, текстур и даже небольшие отличия в дизайне. Эти изменения значительно усложняют задачу сопоставления при использовании более простых методов обработки изображений. Входными данными являются изображения товаров. Эти изображения могут быть получены из различных источников, например, фотографии товаров, изображения из каталогов товаров и т.д. Выходными данными является список идентичных или схожих товаров. Похожие изображения находятся путем сравнения векторов признаков этих изображений. Признаки извлекаются из каждого изображения в вашем наборе данных с помощью обученной нейронной сети. Последний слой сети (перед слоем классификации) используется для извлечения признаков.

В данной работе сравнение векторов изображений товаров производится с помощью нейросетей архитектуры ResNet [11].

Архитектура ResNet состоит из нескольких блоков. Каждый блок содержит несколько сверточных слоев, которые обрабатывают входные данные, и соединение с пропуском, напрямую передающее входные данные на выход блока. Входные данные затем складываются с выходом сверточных слоев, и полученный результат передается следующему блоку. Это позволяет

градиентам обучения пропускать сверточные слои и «проходить» напрямую через соединения с пропуском.

В качестве примера, сеть ResNet-50 состоит из следующих основных компонентов (рис. 1):

1. Сверточный слой (от англ. Convolutional layer): первый сверточный слой применяет 64 фильтра размером 7×7 с шагом 2 (в каждом направлении), после чего следует слой максимального пулинга.

2. Остаточные блоки (от англ. Residual blocks): сеть ResNet-50 имеет всего 16 таких блоков, но каждый из них состоит из 3 слоев вместо 2, что дает общее количество 48 слоев (16 блоков по 3 слоя в блоке). Каждый из этих блоков имеет структуру «bottleneck» (от англ. «горлышко бутылки», слой меньшей размерности), которая включает два сверточных слоя 1×1 по бокам и сверточный слой 3×3 в середине. Это помогает уменьшить количество параметров и ускорить обучение, не жертвуя при этом точностью. Первый слой в каждом блоке уменьшает размерность с использованием свертки 1×1 , второй слой применяет свертки 3×3 , а третий слой восстанавливает размерность обратно с помощью свертки 1×1 . В каждом блоке есть также «shortcut connection» (от англ. «короткие соединения»), позволяющий входным данным обойти эти три слоя.

3. Полносвязный слой (от англ. Fully connected layer): после всех блоков следует полносвязный слой, который приводит 2048 входных единиц к количеству классов, соответствующему задаче классификации.

4. Softmax слой (от англ. Softmax layer): в конце находится слой, реализующий многопеременную логистическую функцию, который выдает вероятностное распределение по классам.

Прежде чем пропустить изображение через сеть архитектуры ResNet, требуется выполнить предварительную обработку, состоящую из следующих шагов:

1. ResNet требует изображения определенного размера. Сети архитектуры ResNet обычно принимают на вход изображения размером 224x224 пикселей, поэтому выполняется приведение к требуемому размеру.

2. Для сети ResNet изображение должно быть нормализовано так, чтобы его пиксели были в диапазоне от 0 до 1, или от -1 до 1. Это достигается путем деления значений пикселей на 255 (если они в диапазоне от 0 до 255).

3. Для улучшения обучения изображения часто центрируются и стандартизируются. Центрирование осуществляется путем вычитания среднего значения пикселей из каждого пикселя. Стандартизация достигается путем деления разности на стандартное отклонение пикселей.

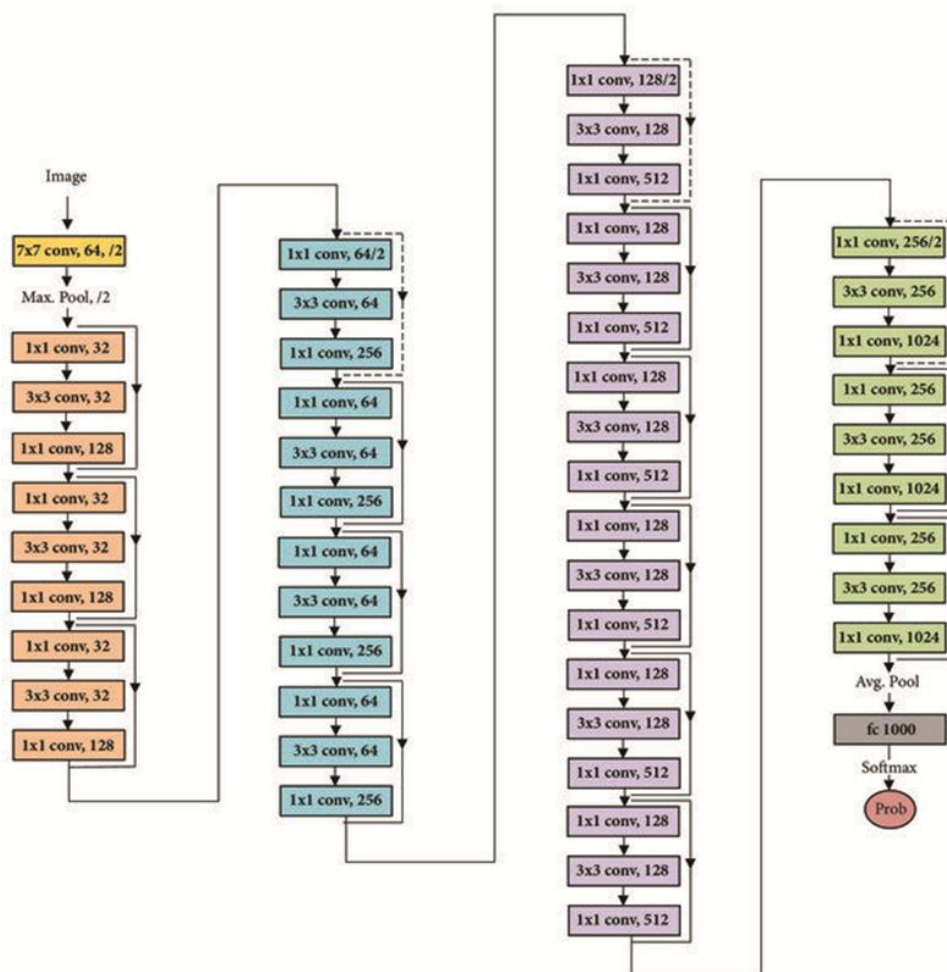


Рис. 1. – Архитектура сети ResNet-50

Каждое изображение трансформируется и нормализуется для модели, после чего извлекаются векторы признаков. Извлечение векторов изображений целесообразно производить с использованием GPU для получения большей производительности. Извлечение векторов признаков является ресурсоёмкой вычислительной задачей, поэтому размер пакета и подбор нужной глубины сети производятся в соответствии с количеством обрабатываемых данных. Результатом является матрица признаков, где каждая строка представляет из себя вектор признаков одного изображения.

В таблице №1 приведено сравнение сетей ResNet по времени извлечения 3000 векторов признаков изображений, при этом размер обрабатываемого пакета равен 32 (`batch_size = 32`). Здесь и далее вычисления производились на машине следующей конфигурации: процессор Intel Core i7-6700K 4.0ГГц, видеокарта NVIDIA GeForce GTX 1050 Ti 4GB, память 16GB DDR3 2133МГц.

Таблица № 1

Время извлечения векторов признаков

Тип использованной сети	Время извлечения, с
ResNet-18	8.5
ResNet-50	17.1
ResNet-101	27.2
ResNet-152	39.2

Для определения схожести между векторами признаков могут быть использованы различные метрики, такие как:

1. Евклидово расстояние (L_2 distance): является одним из наиболее известных и распространенных методов для определения различий между векторами. Представляет собой квадратный корень из суммы квадратов различий между соответствующими элементами двух векторов. Если даны

два вектора $A = (a_1, a_2, \dots, a_n)$ и $B = (b_1, b_2, \dots, b_n)$, то

$$d(A, B) = \sqrt{\sum_{k=1}^n (A_k - B_k)^2}.$$

2. Манхэттенское расстояние (L_1 distance): сумма модулей различий между соответствующими элементами двух векторов. Если есть два вектора $A = (a_1, a_2, \dots, a_n)$ и $B = (b_1, b_2, \dots, b_n)$, то

$$d(A, B) = \sqrt{\sum_{k=1}^n |A_k - B_k|}.$$

3. Косинусное сходство (от англ. Cosine similarity): угол между двумя векторами. Данный метод особенно полезен в ситуациях, где важнее направление векторов, а не их абсолютная величина. Эта мера основана на косинусе угла между двумя векторами и может быть рассчитана по формуле:

$$\text{cosine_similarity} = \cos(\alpha) = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n A_i^2} + \sqrt{\sum_{i=1}^n B_i^2}}.$$

где $A = (a_1, a_2, \dots, a_n)$ и $B = (b_1, b_2, \dots, b_n)$.

4. Сходство Жаккара (от англ. Jaccard similarity): метод измеряет количество общих элементов между двумя наборами. Сходство Жаккара между векторами A и B вычисляется как отношение размера пересечения векторов к размеру их объединения. Данный метод чаще всего используется с наборами, а не векторами, однако может быть адаптирован для работы с векторами:

$$d = \frac{A \cap B}{A \cup B},$$

где $A = (a_1, a_2, \dots, a_n)$ и $B = (b_1, b_2, \dots, b_n)$.

В общем случае, при работе с векторами признаков изображений, которые обычно являются высокоразмерными и разреженными, косинусное сходство часто оказывается более эффективным. Косинусное сходство

учитывает угол между векторами, а не модуль суммы различий между соответствующими элементами двух векторов, что дает достойные результаты в случае сравнения векторов признаков изображений. Также преимуществом является то, что результат нормализован, то есть находится в диапазоне от -1 до 1. Значение 1 означает, что векторы сонаправленные, а значение -1 означает, что векторы направлены в противоположные стороны, таким образом, в контексте сравнения изображений, если косинус угла между их векторами признаков близок к 1, то изображения являются похожими.

Для сравнения двух наборов изображений используется матричное умножение массивов векторов признаков. Это позволяет вычислить сходство между всеми парами векторов признаков из двух наборов изображений. В контексте вычисления сходства между векторами признаков, матричное умножение используется для вычисления скалярного произведения между каждым вектором признаков из одного набора и каждым вектором признаков из другого набора.

Таким образом, пусть X — матрица признаков первого набора изображений размера (n_1, d) , а Y — матрица признаков второго набора изображений размера (n_2, d) , где n_1, n_2 — количество изображений в первом и втором наборах соответственно, а d — размерность векторов признаков.

Тогда матрица сходства S размера $n_1 \times n_2$ вычисляется, как:

$$S = X_{norm} \times Y_{norm},$$

где X_{norm}, Y_{norm} — нормализованные матрицы X, Y соответственно.

Элемент $S[i, j]$ представляет собой степень сходства между i -м вектором признаков из первого набора и j -м вектором признаков из второго набора.

Матричное умножение для вычисления сходства векторов признаков используется вместо поэлементного сравнения, поскольку поддается

параллелизму, то есть множество вычислений может осуществляться одновременно, например, с применением GPU, что позволяет повысить эффективность за счет уменьшения времени сравнения. Современные библиотеки линейной алгебры и фреймворки для глубокого обучения, такие как PyTorch, оптимизированы для выполнения матричного умножения на CPU и GPU [12]. В таблице №2 приведено время поэлементного и матричного сравнения векторов признаков 3000 изображений с 100 изображениями.

Таблица №2

Время сравнения векторов признаков

Тип сравнения	Время сравнения, с
Поэлементное сравнение	25.4
Матричное сравнение	0.7

При матричном умножении данные могут быть загружены в память и обработаны блоками, что уменьшает количество операций ввода-вывода и пересылки данных между различными уровнями памяти (например, пересылка векторов с GPU на CPU). После проведения вычислений осуществляется проход по строкам полученной матрицы, и для каждого вектора признаков из первого набора выполняется поиск индексов векторов из второго набора, сходство с которыми превышает определенный порог:

$$S[i, j] > n,$$

где n — порог совпадения.

Таким образом, становится возможным выяснить, какие изображения из второго набора являются наиболее похожими на каждое изображение из первого набора.

Оценка эффективности сопоставления.

Величина порога совпадения косинусного сходства, который определяет, насколько два вектора считаются схожими, обычно подбирается эмпирически и может значительно варьироваться в зависимости от специфики задачи, набора данных и архитектуры нейронной сети. Как было отмечено выше, косинусное сходство варьируется в диапазоне от -1 (полное несоответствие) до 1 (полное соответствие), где значение, близкое к 1, указывает на высокую степень сходства между векторами.

При выборе порога сходства следует учитывать следующее:

1. Зависимость от контекста: для разных типов данных и задач одно и то же значение порога может интерпретироваться по-разному. Например, в задаче нахождения повторов(дубликатов) изображений значение 0.975 может считаться нижней границей, в то время как в контексте поиска похожих изображений — достаточно высоким значением.

2. Влияние архитектуры сети: разные архитектуры нейросетей могут по-разному извлекать и интерпретировать признаки из данных, что влияет на распределение косинусных сходств векторов признаков. Следовательно, оптимальный порог сходства может отличаться для различных сетей.

3. Эмпирический подбор: часто определение порога сходства требует экспериментальной оценки с использованием набора размеченных данных, соответствующих задаче.

Для наглядной демонстрации работы разработанной системы были выбраны несколько пар изображений товаров из разных торговых сетей. Так, например, в контексте сравнения векторов признаков сети ResNet-50 и изображений товаров, значение косинусного сходства около 0.9 означает, что изображения похожи друг на друга. Они могут иметь малое количество заметных различий, но большинство характеристик изображений совпадают. На рис. 2 приводится пример умеренной схожести двух изображений,

косинусное сходство при этом равно 0.9265738725662231, применяемая модель сети — ResNet-50.



Рис. 2. – Пример умеренной степени схожести векторов признаков.

На рис. 3 также изображен пример умеренной схожести для двух изображений, косинусное сходство 0.9185383231985474, модель сети — ResNet-50.



Рис. 3. – Пример умеренной степени схожести векторов признаков.

Пример на рис. 3 двух изображений, отличающихся цветовой гаммой, показывает, что нейросеть устойчива к изменению цвета товаров и определяет эти товары как похожие. Косинусное сходство от 0.95 до 0.99 указывает на высокую степень схожести, близкую к идентичности.

На рис. 4 приведен пример двух изображений, которые сеть определяет как изображения с высокой степенью схожести, косинусное сходство 0.9619644379615784, модель сети — ResNet-50:



Рис. 4. – Пример высокой степени схожести векторов признаков

Данные изображения отличаются только первоначальными размерами, что показывает, что сеть хорошо справляется с изменениями размера изображений.

Косинусное сходство от 0.99 до 1 означает совпадение векторов признаков и, следовательно, идентичность изображений. Они либо полностью совпадают, либо любые различия между ними настолько незначительны, что они не влияют на общие характеристики изображений. Пример (рис. 5) двух изображений, отличающихся только первоначальными размерами, косинусное сходство 0.990291953086853, модель сети — ResNet-50.



Рис. 5. – Пример совпадения векторов признаков

При значениях очень близких (в пределах от 0.999 до 1) или равных 1 можно утверждать о полной идентичности изображений, в том числе исходных размеров изображений. При значениях сходства менее 0.9

наблюдаются товары, которые могут быть схожи по внешним признакам (например, упаковка товара в бутылированном виде), но сильно отличаются по цветовой гамме и форме объекта.

Для экспериментов использовались изображения товаров из двух крупных торговых сетей. Все изображения были приведены к единому размеру 224×224 пикселя для соответствия входным требованиям архитектуры ResNet. Нормализация изображений проводилась путем деления значений пикселей на 255. Центрирование осуществлялось путем вычитания среднего значения пикселей из каждого пикселя изображения, а стандартизация — делением на стандартное отклонение.

Для оценки эффективности сопоставления были использованы следующие метрики [13]:

- Recall (Полнота) — доля правильно найденных положительных объектов среди всех объектов положительного класса:

$$Recall = \frac{TP}{TP + FN},$$

где TP — правильно найденные объекты, FN — объекты с ложно предсказанной отрицательной меткой. Далее в работе метрика Recall именуется как «совпадения» и выражается в процентах.

- Ложноположительные — доля неправильно найденных положительных объектов среди всех правильно найденных положительных объектов:

$$Ложноположительные = \frac{FP}{TP},$$

где FP — найденные ложноположительные объекты, TP — правильно найденные объекты. Далее выражается в процентах.

В таблице 3 представлены результаты сопоставления 1000 изображений товаров из двух торговых сетей с использованием трех архитектур нейронных сетей ResNet: ResNet-18, ResNet-50 и ResNet-152 для разных порогов схожести. Изображения могут как значительно различаться

по своим визуальным характеристикам (вид упаковки, цвет, некоторые различия в дизайне), так и могут быть полностью идентичными. Такой набор данных обеспечивает достаточный объем информации для тестирования устойчивости и точности сопоставления изображений.

Таблица 3.

Точность сопоставления изображений товаров

Порог схожести	Результаты для ResNet-18		Результаты для ResNet-50		Результаты для ResNet-152	
	совпадения, %	ложно-положительные, %	совпадения, %	ложно-положительные, %	совпадения, %	ложно-положительные, %
0.85	59.54	28.6	60.85	29.62	61.24	29.28
0.90	48.21	9.84	51.55	9.74	53.61	9.93
0.95	37.13	4.82	39.81	4.54	42.68	4.69
0.99	11.11	1.24	11.38	1.18	11.34	1.12

Результаты, представленные в таблице 3, демонстрируют долю найденных совпадений между тестовыми изображениями торговых сетей и долю ложноположительных результатов для каждой из архитектур при разных порогах схожести.

Выводы

В ходе работы были протестированы нейронные сети архитектуры ResNet для сопоставления изображений товаров. Как и следовало ожидать, полученные результаты показывают, что с увеличением сложности модели (переход от ResNet-18 к ResNet-152) наблюдается рост точности сопоставления изображений товаров. Однако при этом более сложные модели являются «громоздкими», и сопоставление занимает больше времени. Следует отметить, что выбор оптимальной архитектуры сети для применения в конкретной компании является самостоятельной задачей, решение которой зависит от ряда факторов, в том числе от доступных ресурсов, а также

ограничений по времени операции сопоставления. Среди направлений дальнейших исследований следует отметить возможность совмещения результатов сопоставления товаров по изображениям и обработки текстовой информации. Кроме того, имеет смысл рассмотреть возможность дообучения модели нейронной сети на задаче уменьшения расстояния между векторами признаков изображений одинаковых товаров.

Литература

1. «Лента» тестирует технологии видеораспознавания товаров и количества покупателей в магазине. URL: lenta.com/o-kompanii/news/-----30. (дата обращения: 17.07.2024).
2. «Магнит» до конца года масштабирует технологию распознавания товаров на полке на всю сеть. URL: dsmedia.pro/news/magnit-do-konca-goda-masshtabiruet-tehnologiju-raspoznavanija-tovarov-na-polke-na-vsju-set. (дата обращения: 17.07.2024).
3. Wahyu W.W. Searching Similarity Digital Image Using Color Histogram / W.W. Wahyu, K. K. Kusriani, A.F. Hanif // Techno (Jurnal Fakultas Teknik Universitas Muhammadiyah Purwokerto). – 2019. – V. 20, № 1. – pp. 53-64. – ISSN 1410-8607
4. Shahrin M., Jain R., Smith T. Analytical Comparison of Histogram Distance Measures // Journal of Image Processing. – 2019. – V. 8. – № 2. – pp. 123-135.
5. Рудаков И. В., Васютович. И. М. Исследование перцептивных хеш-функций изображений // Машиностроение и компьютерные технологии. – 2015. – № 8. – С. 269-280.
6. Алексеев В.В, Шоберг А.Г. Частотное разделение характеристик изображения в задачах сверхразрешения // Инженерный вестник Дона, 2024, №8. URL: ivdon.ru/ru/magazine/archive/n8y2024/9375 (дата обращения: 17.07.2024).

7. Ковалев А.В., Исаева А.С. Оценка качества семян пшеницы с использованием сверточной нейронной сети // Инженерный вестник Дона, 2021, №12. URL: ivdon.ru/ru/magazine/archive/n12y2021/7354 (дата обращения: 17.07.2024).

8. Сикорский О.С. Обзор свёрточных нейронных сетей для задачи классификации изображений // Новые информационные технологии в автоматизированных системах. – 2017. – № 20. – С. 37-42. – ISSN 2227-0973

9. Appalaraju S., Chaoji V. Image similarity using Deep CNN and Curriculum Learning // arXiv. – 2017. – URL: doi.org/10.48550/arXiv.1709.08761 (дата обращения: 17.07.2024).

10. Deep Residual Learning for Image Recognition. URL: arxiv.org/abs/1512.03385 (дата обращения 17.07.2024).

11. Прокопеня А.С. Сверточные нейронные сети для распознавания изображений // BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня: сб. материалов VI Междунар. науч.-практ. конф., Минск, 20-21 мая 2020 года: в 3 ч. Ч. 1 / редкол. : В. А. Богуш [и др.]. – Минск : Бестпринт, 2020. – С. 271–280.

12. Документация PyTorch. URL: pytorch.org/ (дата обращения: 17.07.2024).

13. Гладилин П.Е., Боченина К.О., Технологии машинного обучения // СПб: Университет ИТМО, 2020. – 75 с.

References

1. «Lenta» testiruet tekhnologii videoraspoznavaniya tovarov i kolichestva pokupatelej v magazine [The Lenta company is testing technologies for video recognition of goods and the number of customers in the store]. URL: lenta.com/o-kompanii/news/-----30. (date accessed: 17.07.2024).

2. «Magnit» do konca goda masshtabiruet tekhnologiyu raspoznavaniya tovarov na polke na vsyu set [The Magnit company will scale the technology of

recognizing goods on the store shelf to the entire network by the end of the year].

URL: dsmedia.pro/news/magnit-do-konca-goda-masshtabiruet-tehnologiju-raspoznaniya-tovarov-na-polke-na-vsju-set. (date accessed: 17.07.2024).

3. W.W. Wahyu, K. K. Kusriani, A.F. Hanif. Techno (Jurnal Fakultas Teknik Universitas Muhammadiyah Purwokerto). 2019. V. 20, № 1. pp. 53-64. ISSN 1410-8607

4. Shahrin M., Jain R., Smith T. – Journal of Image Processing. 2019. V. 8, № 2, pp. 123-135.

5. I. V. Rudakov, I. M. Vasyutovich. Mashinostroenie i komp'yuternye tekhnologii. 2015. № 8, pp. 269-280. ISSN 2587-9278.

6. Alekseev V.V, Shoberg A.G. Inzhenernyj vestnik Dona, 2024, №8. URL: ivdon.ru/ru/magazine/archive/n8y2024/9375 (date accessed: 17.07.2024).

7. Kovalev A.V., Isaeva A.S. Inzhenernyj vestnik Dona, 2021, №12 URL: ivdon.ru/ru/magazine/archive/n12y2021/7354 (date accessed: 17.07.2024).

8. Sikorskij O.S. Novye informacionnye tekhnologii v avtomatizirovannyh sistemah. 2017. № 20, pp. 37-42. ISSN 2227-0973.

9. Appalaraju S., Chaoji V. arXiv. 2017. URL: doi.org/10.48550/arXiv.1709.08761 (date accessed 17.07.2024).

10. Deep Residual Learning for Image Recognition. URL: arxiv.org/abs/1512.03385 (date accessed: 17.07.2024).

11. Prokopenya A.S. BIG DATA and Advanced Analytics = BIG DATA i analiz vysokogo urovnya: sb. materialov VI Mezhdunar. nauch.-prakt. konf., Minsk, 20-21 maya 2020 goda: Part 1. Redkol. : V. A. Bogush [i dr.]. Minsk: Bestprint, 2020. pp. 271–280.

12. Dokumentaciya PyTorch [PyTorch documentation]. URL: pytorch.org/ (date accessed: 17.07.2024).

13. Gladilin P.E., Bochenina K.O., Tekhnologii mashinnogo obucheniya [Machine learning technologies]. Spb: Universitet ITMO. 2020. 75 p.

Дата поступления: 28.06.2024

Дата публикации: 8.08.2024
