

Применение сверточных нейронных сетей для распознавания объектов на изображении

А. И. Волгушева, Р. В. Мальчева

Email: korkoalena@yandex.ru, raisa.malcheva@yandex.ru

Аннотация:

Выполнен анализ принципов работы искусственных нейронных сетей, сферы их применения для распознавания объектов на изображении. Выбран и реализован алгоритм на основе сверточных нейронных сетей, способный с довольно высокой точностью обнаруживать на изображении выбранные классы объектов. Использование данного алгоритма для анализа видеопотоков, получаемых с касс самостоятельного обслуживания, может позволить предприятию, такому как зал кафе или ресторан быстрого питания, функционировать без участия человека.

Введение

В последнее время области применения машинного зрения быстро расширяются, охватывая все больше различных сфер жизнедеятельности. На сегодняшний день для реализации систем машинного зрения в основном применяются сверточные нейронные сети, которые являются одним из видов искусственных нейронных сетей. Именно благодаря способности к обучению, искусственные нейронные сети используются для решения сложных задач классификации и регрессионного анализа.

Системы компьютерного зрения, использующие такой подход, обладают хорошей устойчивостью к изменениям масштаба, смещениям, поворотам, смене ракурса и прочим искажениям. На основе сверточных нейронных сетей реализовываются системы для беспилотного управления транспортными средствами, распознавания лиц и выявления различных заболеваний на медицинских снимках.

Актуальность данной работы обуславливается тем, что применение методов распознавания объектов на изображении может помочь уменьшить роль человеческого фактора на предприятии. Одним из таких случаев являются столовые и рестораны быстрого питания.

Целью данной работы является реализация алгоритма, способного распознавать заданные объекты на изображении.

Обзор существующих методов распознавания объектов на изображении

В рамках данной работы требуется разработать алгоритм, способный распознавать объекты на изображении, относящиеся к категориям продуктов и блюд. Визуально блюдо

может представляться как один объект из этих категорий, так и комбинацией двух объектов. Процесс распознавания объекта на изображении включает два этапа: локализацию объекта на изображении и присвоение ему соответствующей категории на основе его свойств.

В решении задачи распознавания объекта на изображении применяются различные методы, включая наложение цветowych фильтров, выделение и анализ контуров, сопоставление с шаблонами, а также поиск особых точек. Однако эти методы не подходят для распознавания блюд, так как они обладают разнообразной формой и могут находиться в различных положениях на изображении. Поэтому для достижения цели работы было принято решение использовать методы машинного обучения.

Машинное обучение является разделом искусственного интеллекта, который занимается разработкой систем, способных решать задачи, решение которых ранее считалось прерогативой человека. Наиболее популярным методом идентификации объектов на изображении являются искусственные нейронные сети, особенно сверточные нейронные сети.

Эти сети обладают частичной устойчивостью к изменениям в масштабе, смещениям, поворотам, смене ракурса и прочим искажениям.

Однако, недостатком сверточных нейронных сетей является их неспособность классифицировать множество объектов на одном изображении, что является необходимым для распознавания блюд.

Для решения таких задач разработаны специальные алгоритмы на основе сверточных нейронных сетей, такие как R-CNN, YOLO и SSD, которые позволяют локализовать и классифицировать множество объектов на изображении.

Анализ принципа работы искусственной нейронной сети

Искусственная нейронная сеть – это математическая модель, построенная на основе концепции функционирования мозга. Она используется для решения разнообразных практических задач. Нейронная сеть имитирует работу нейронов в мозге, которые представляют собой простые вычислительные элементы, обменивающиеся сигналами друг с другом и способные выполнять сложные операции. Искусственные нейроны представляют функцию, которой подаются входные значения, умноженные на регулируемые веса. Затем происходит суммирование результатов и полученное значение проходит через нелинейную функцию, определяющую степень возбуждения нейрона. Обучаемость и нелинейность являются ключевыми характеристиками искусственных нейронных сетей, которые позволяют находить сложные связи между данными и учитывать опыт для улучшения результатов работы [1-12].

Структура искусственной нейронной сети представляет собой направленный граф, в котором нейроны выступают в роли узлов, а связи с весовыми коэффициентами являются дугами. Указанная структура показана на рис. 1.

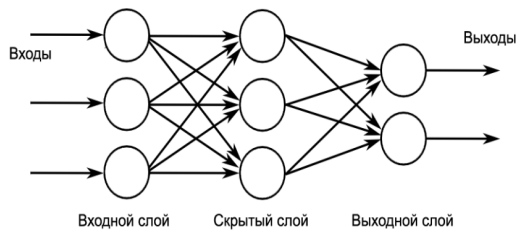


Рисунок 1 – Структура искусственной нейронной сети

Понять принцип работы искусственной нейронной сети можно на примере персептрона – одной из первых моделей нейронной сети. Модель персептрона представлена на рис. 2.

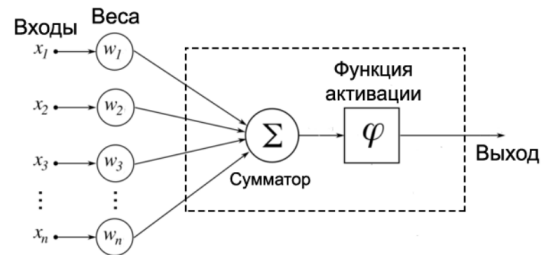


Рисунок 2 – Модель персептрона

Между входными нейронами и сумматором существуют связи с весовыми коэффициентами, которые определяют вклад каждого входа в формирование выходного значения. Сумматор вычисляет взвешенную сумму входных сигналов, умноженных на их соответствующие веса. Эта сумма затем передается в функцию активации, которая определяет активацию нейрона.

Функция активации производит нормализацию сигнала и приводит его к нужному диапазону в зависимости от типа функции активации. Без использования функции активации нейронная сеть превращается в обычную модель линейной регрессии, поскольку именно она позволяет выполнять нелинейное преобразование данных и решать более сложные задачи. Существует множество различных видов функций активации, включая ступенчатую, линейную, сигмоидальную, гиперболический тангенс и функцию усеченного линейного преобразования (ReLU), которые представлены на рис. 3.

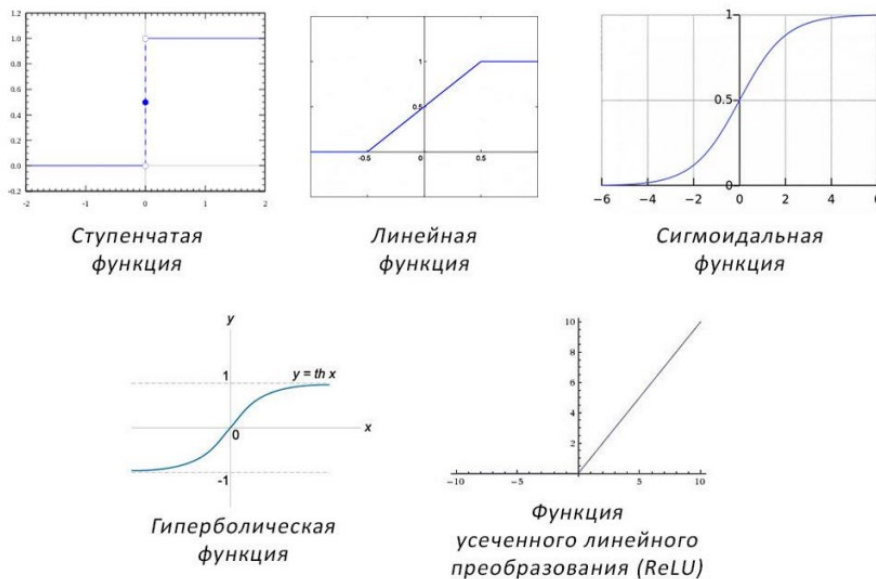


Рисунок 3 – Основные функции активации

При выборе функции активации для нейронной сети следует учитывать тип задачи, которую необходимо решить. Персептрон, самый простой вид нейронной сети, подходит для простых задач классификации, но не может справиться с более сложными задачами. Именно поэтому были разработаны многослойные нейронные сети прямого распространения, которые состоят из нейронов, организованных в слои. Нейроны каждого слоя связаны с нейронами следующего слоя, но не соединены между собой. Все слои, кроме первого и последнего, называются скрытыми слоями.

В большинстве архитектур нейронных сетей часто используется смещение во входных и скрытых слоях - нейрон, который не принимает входных данных, так как его входной сигнал всегда равен единице, однако позволяет сдвигать функцию активации в разные направления, делая модель более гибкой. Нейронная сеть с многослойными нейронами, включая нейроны смещения, изображена на рис. 4.

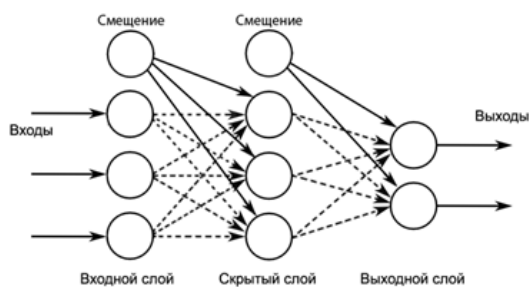


Рисунок 4 – Многослойная нейронная сеть с нейронами смещения

Существует большое число различных архитектур нейронных сетей, отличающихся количеством скрытых слоев, нейронов и видами связей между ними, каждая из которых лучше справляется со своим типом задач, однако, для того чтобы нейронная сеть научилась решать задачи разного рода, её необходимо обучать.

Особенности сверточной нейронной сети

Сверточная нейронная сеть является одной из наиболее популярных архитектур нейронных сетей, применяемая для анализа визуальных образов и классификации изображений. При восприятии различных образов, в зрительной коре активируются разные группы нейронов. В целом, архитектура сверточной нейронной сети позволяет выделять в данных различные признаки, начиная от очень простых и заканчивая более сложными.

Свёрточная нейронная сеть состоит из чередующихся слоев свертки и подвыборки, после которых следуют полносвязные слои. Архитектура свёрточной нейронной сети представлена на рисунке 5. Свёрточный слой всегда является первым в сверточной нейронной сети и реализует операцию свёртки, которая отражает реакцию отдельного нейрона на конкретную область поля зрения.

На вход свёрточного слоя подается изображение в виде матрицы пикселей, размерность которой определяется размером изображения. Цветные изображения представляются тремя матрицами, каждая из которых отвечает за один из каналов цветовой модели RGB, значения матриц описывают интенсивность цвета пикселей.

В качестве нейрона выступает некое ядро свертки – матрица, элементы которой являются весовыми коэффициентами [12].

Глубина ядра свертки должна быть такой же, как и глубина изображения, к примеру для цветного изображения применяется ядро свертки, состоящее из трех матриц, каждая из которых применяется для свертки по одному из каналов цветного изображения. Операция свертки начинается с определения рецептивного поля, размер которого равен ядру свертки.

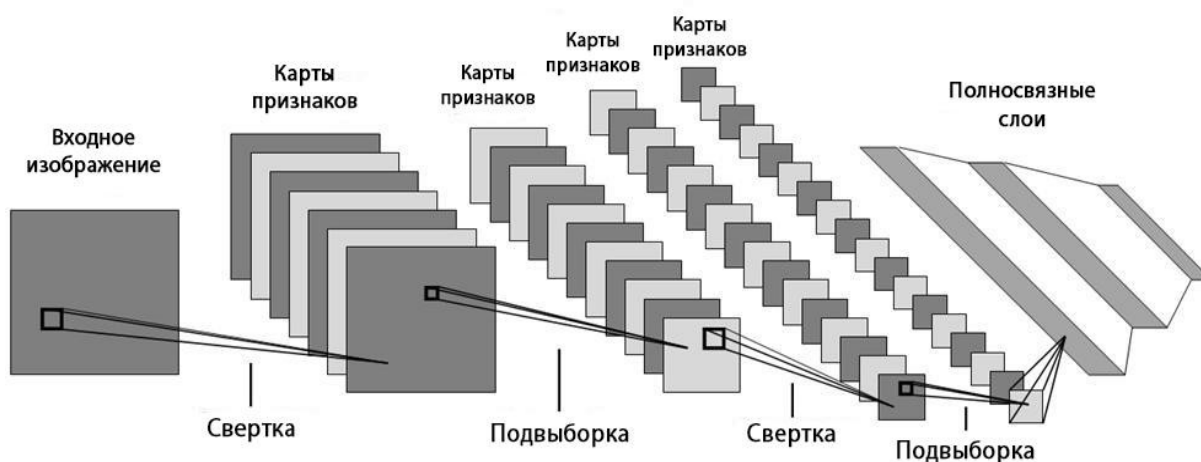


Рисунок 5 – Архитектура свёрточной нейронной сети

Данное поле скользит с некоторым смещением по исходному изображению и пиксели изображения, попавшие в него, поэлементно перемножаются с ядром свертки. Элементы новой матрицы, полученной в результате перемножения, суммируются и образуют новый выходной пиксель. Операция свертки для черно-белого изображения представлена на рисунке 6.

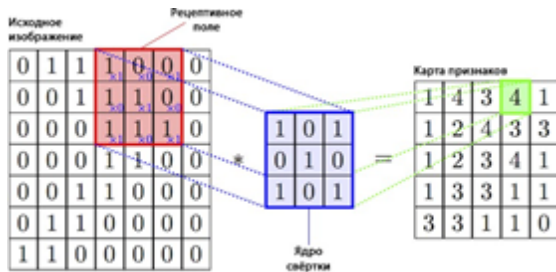


Рисунок 6 – Операция свертки

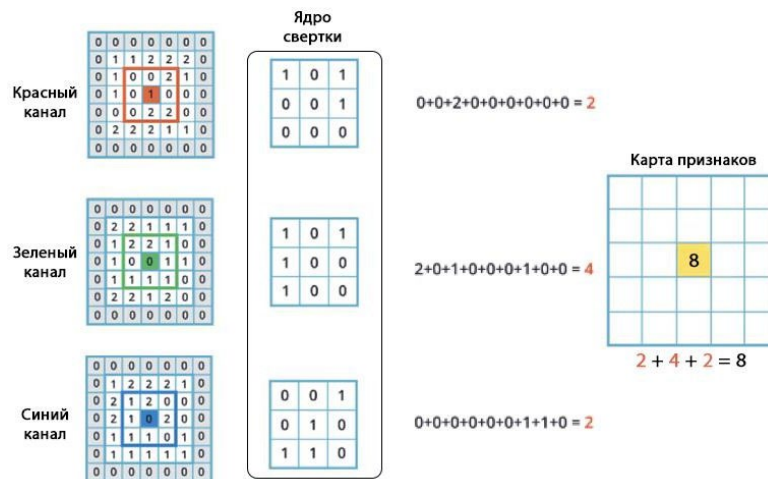


Рисунок 7 – Свертка цветного изображения с применением заполнения

Также, как и в многослойных нейронных сетях, в данной архитектуре присутствует смещение – значение, изменяемое в процессе обучения, которое складывается с каждым элементом выходной матрицы. Перед поступлением на следующий слой, полученная матрица проходит через функцию активации. Обычно в сверточных слоях в качестве функции активации используют функцию усеченного линейного преобразования, которая преобразовывает все отрицательные значения в 0, а остальные не изменяет. Данная функция была представлена на рисунке 1.3.

Итак, в результате свертки получается матрица, называемая картой признаков, на которой выделен конкретный признаки, такой как: различные прямые и кривые линии, полуокружности или более сложные признаки, в зависимости местонахождения сверточного слоя в модели нейронной сети.

Свертка цветного изображения отличается лишь тем, что к каждому цветовому каналу применяется одна из матриц, составляющих трехмерное ядро свертки, после чего полученные матрицы суммируются в одну результирующую матрицу. Из-за того, что граничные пиксели исходного изображения не оказываются в центре рецептивного поля, матрица, полученная в результате свертки, имеет меньшую размерность, чем исходное изображение. Чтобы избежать постоянного сжатия изображения при выполнении свертки и сохранить информацию о его границах, вокруг изображения добавляется рамка из нулей, ширина которой зависит от размера ядра свертки. Данная операция называется заполнением (zero-padding). Свертка цветного изображения с заполнением представлена на рисунке 7.

После слоя свертки обычно следует слой подвыборки. Данный слой уменьшает пространственную размерность карты признаков, это достигается путем замены некоторой области пикселей одним пикселем. Необходимость этой операции объясняется тем, что важна именно информация о наличии конкретных признаков на изображении, а не их местоположение, поэтому оно сжимается до менее подробного, состоящего из доминирующих признаков [12]. Кроме того, операция подвыборки уменьшает количество параметров сверточной сети, тем самым снижая вычислительную нагрузку.

При выполнении операции подвыборки учитываются два параметра: размер ядра подвыборки, скользящего по изображению, которое сворачивает попавшие в него пиксели до 1 пикселя и размер шага в пикселях, на которое сдвигается ядро подвыборки. Чаще всего используют пара значений параметров: 2x2,

которая уменьшает размерность карты признаков в два раза.

Для того, чтобы выявить сложные признаки на изображении, сверточные сети содержат множество чередующихся слоев свертки и подвыборки. Таким образом, с каждой парой слоев свертки глубина карты признаков изменяется в зависимости от числа ядер свертки, а их размерность уменьшается.

Конечными слоями сверточной нейронной сети являются полносвязные слои, с помощью которых происходит классификация объекта на изображении. Перед подачей полученной карты признаков на полносвязный слой, происходит объединение слоев карты признаков в вектор, где каждый элемент показывает вероятность присутствия конкретного признака на изображении. К примеру, чтобы изображение было классифицировано как лицо, вероятность присутствия таких признаков как глаза, рот и нос должна быть высокой [10]. Последний полносвязный слой содержит количество нейронов, равное количеству определяемых классов и с помощью выбранной функции активации выводит вероятности присутствия каждого класса на изображении.

Реализация алгоритма ssd для локализации возгораний

Для реализации алгоритма SSD был выбран высокоуровневый кроссплатформенный объектно-ориентированный язык программирования Python. Данный язык программирования является мощным инструментом, так как содержит множество библиотек, реализующих программные решения для разработки различных систем, в частности для решения задач машинного обучения [5].

Для реализации и обучения нейронной сети, используемой в алгоритме SSD, была выбрана библиотека машинного обучения TensorFlow версии 2.0. Она включает в себя множество готовых решений и алгоритмов, таких как функции оптимизации, функции потерь, различные метрики, а также инструменты для проектирования различных нейронных сетей [10]. Для работы с данной библиотекой все обрабатываемые ею данные представляются в виде специальных контейнеров, называемых тензорами – многомерными массивами. На данный момент TensorFlow является наиболее популярной библиотекой для машинного обучения.

В первую очередь при реализации алгоритма для распознавания блюд на основе архитектуры SSD были реализованы сверточные слои, используемые в модели нейронной сети. Построение слоев модели выполняется с помощью встроенного в Tensorflow

высокоуровневого API для построения моделей под названием Keras.

Создание сверточных слоев происходит с помощью метода «Conv2D», который в качестве параметров принимает количество ядер свертки, их размерность, размер граничного заполнения, расширение и функцию активации, используемую после свертки. Для создания слоев подвыборки по максимальному значению используется метод «MaxPool2D», принимающий аналогичные параметры за исключением расширения и функции активации. Для задачи входных данных нейронной сети используется метод Input, который создает входной тензор определенного размера. Помимо входного тензора в данную функцию необходимо передать выходные данные последнего слоя. Программный код реализации слоев сверточной сети VGG16 представлен на рисунке 8.

```
def create_vgg16_layers():
    vgg16_conv4 = [
        layers.Conv2D(64, 3, padding='same', activation='relu'),
        layers.Conv2D(64, 3, padding='same', activation='relu'),
        layers.MaxPool2D(2, 2, padding='same'),

        layers.Conv2D(128, 3, padding='same', activation='relu'),
        layers.Conv2D(128, 3, padding='same', activation='relu'),
        layers.MaxPool2D(2, 2, padding='same'),

        layers.Conv2D(256, 3, padding='same', activation='relu'),
        layers.Conv2D(256, 3, padding='same', activation='relu'),
        layers.Conv2D(256, 3, padding='same', activation='relu'),
        layers.MaxPool2D(2, 2, padding='same'),

        layers.Conv2D(512, 3, padding='same', activation='relu'),
        layers.Conv2D(512, 3, padding='same', activation='relu'),
        layers.Conv2D(512, 3, padding='same', activation='relu'),
        layers.MaxPool2D(2, 2, padding='same'),

        layers.Conv2D(512, 3, padding='same', activation='relu'),
        layers.Conv2D(512, 3, padding='same', activation='relu'),
        layers.Conv2D(512, 3, padding='same', activation='relu'),

        layers.MaxPool2D(3, 1, padding='same'),
        layers.Conv2D(1024, 3, padding='same', dilation_rate=6, activation='relu'),
        layers.Conv2D(1024, 1, padding='same', activation='relu'),
    ]

    vgg16_conv7 = [
        layers.MaxPool2D(3, 1, padding='same'),
        layers.Conv2D(1024, 3, padding='same', dilation_rate=6, activation='relu'),
        layers.Conv2D(1024, 1, padding='same', activation='relu'),
    ]
```

Рисунок 8 - Программный код реализации слоев сверточной сети VGG16

Дополнительные слои, а также слои для классификации и регрессии реализованы таким же образом и будут приведены в приложении. Для того, чтобы использовать эти слои при прямом проходе, был создан класс для модели SSD. Данный класс содержит метод инициализации модели, в котором загружаются веса, настроенные в процессе долгого обучения на наборе данных ImageNet, содержащим порядка 15 миллионов размеченных изображений, разделенных на десятки тысяч классов. Данный подход позволяет ускорить процесс обучения рассматриваемой архитектуры для распознавания новых классов объектов.

Программный код функции, осуществляющей прямой проход в модели SSD представлен на рисунке 9.

```
def call(self, x):  
    """ Прямой проход  
    """  
    confs = []  
    locs = []  
    head_idx = 0  
    for i in range(len(self.vgg16_conv4.layers)):  
        x = self.vgg16_conv4.get_layer(index=i)(x)  
        if i == len(self.vgg16_conv4.layers) - 5:  
            conf, loc = self.compute_heads(self.batch_norm(x), head_idx)  
            confs.append(conf)  
            locs.append(loc)  
            head_idx += 1  
  
    x = self.vgg16_conv7(x)  
  
    conf, loc = self.compute_heads(x, head_idx)  
  
    confs.append(conf)  
    locs.append(loc)  
    head_idx += 1  
  
    for layer in self.extra_layers:  
        x = layer(x)  
        conf, loc = self.compute_heads(x, head_idx)  
        confs.append(conf)  
        locs.append(loc)  
        head_idx += 1  
  
    confs = tf.concat(confs, axis=1)  
    locs = tf.concat(locs, axis=1)  
  
    return confs, locs
```

Рисунок 9 - Программный код функции, осуществляющей прямой проход в модели SSD

Данная функция принимает на вход изображение и совершает прямой проход для предсказания ограничивающих рамок объектов на изображении и их классов.

Генерация наборов обучающий данных происходит с помощью функции «create_batch_generator», предоставляющей объекты класса генератора тренировочных и проверочных данных. Эти объекты-генераторы позволяют в процессе обучения получать наборы данных, состоящие из изображений, меток классов для всех рамок по умолчанию, а также смещений для положительных рамок по умолчанию относительно реальных рамок объекта.

Метод генерации использует специальный класс «VOCDataset», который подготавливает данные для обучения: считывает из обучающего набора данных тренировочные и проверочные изображения, а также метки классов с ограниченными прямоугольниками, изменяет размеры изображения и находит смещения и метки классов для положительных рамок по умолчанию [7]. Создание генератора обучающих наборов данных происходит с помощью метода «Dataset.from_generator» библиотеки TensorFlow.

Для автоматического дифференцирования обучаемых переменных при обратном распространении ошибки, необходимо

использовать специальный диспетчер контекста «tf.GradientTape()». Внутри данного контекста используется функция для расчёта ошибок модели SSD. Получаемые с помощью неё ошибки классификации и локализации суммируются, образуя общую ошибку детектора для одного обучающего примера [4]. После этого находится сумма квадратов ошибок для всех примеров обучающего пакета и делится на их количество. Общая ошибка используется для нахождения градиента, который используется методом оптимизации для уменьшения ошибки обучаемой модели. В качестве оптимизатора используется стохастический градиентный спуск, который задается с помощью функции библиотеки TensorFlow «SDG». В качестве параметров метода оптимизации передается значение скорости обучения и значения импульса, который используется для учета уровня влияния градиентов, рассчитанных на предыдущих шагах обучения. Данный параметр позволяет обновлять веса более сглажено, что способствует более быстрой оптимизации ошибки модели. Рекомендуется использовать импульс равный 0,9 [11].

Заключение

В данной работе исследовался вопрос распознавания блюд на изображении на основе сверточных нейронных сетей.

Проведен анализ существующих методов распознавания объектов на изображении и сделан вывод, что для распознавания блюд наиболее подходящими являются методы, основанные на машинном обучении, а именно сверточные нейронные сети. Рассмотрены основные принципы работы искусственных нейронных сетей, особенности их построения, а также методы, с помощью которых происходит их обучение. Кроме того, подробно разобран процесс применения сверточных нейронных сетей для классификации объекта на изображении.

Для локализации и классификации объектов на изображении принято решение использовать алгоритм Single Shot MultiBox Detector, в основу которого входят сверточные нейронные сети. Данный алгоритм выбран, поскольку он обнаруживает объекты за один прямой проход нейронной сети и достигает наилучшего баланса между точностью и быстродействием, что является важным при распознавании. Также описаны основные принципы работы алгоритма SSD и произведена его реализация на языке Python с помощью инструментов библиотеки машинного обучения TensorFlow.

Для обучения модели сверточной нейронной сети, входящей в состав алгоритма SSD, вручную собран и размечен обучающий

набор данных, состоящий из 2000 изображений. Обучение сверточной нейронной сети проходило в течение 300 эпох, в результате чего минимальная средняя ошибка предсказания составила 0,4.

Таким образом, реализован алгоритм на основе сверточных нейронных сетей, способный обнаруживать на изображении выбранные классы объектов, с довольно высокой точностью. Использование данного алгоритма для анализа видеопотоков, получаемых с касс самостоятельного обслуживания (КСО), может позволить предприятию (залу кафе или ресторану быстрого питания) функционировать без участия человека.

Литература

1. Бурков, А. Машинное обучение без лишних слов / А. Бурков. – СПб.: Питер, 2020. – 192 с.
2. Вьюгин, В. Математические основы машинного обучения и прогнозирования / В. Вьюгин. - МЦНМО, 2014. - 304 с.
3. Гелиг, А. Введение в математическую теорию обучаемых распознающих систем и нейронных сетей. Учебное пособие / А. Гелиг, А. Матвеев - Издательство СПбГУ, 2014. – 224 с.
4. Документация по библиотеке машинного обучения TensorFlow 2.0: официальный сайт TensorFlow [Электронный ресурс]. - URL: https://www.tensorflow.org/versions/r2.0/api_docs/python/
5. Документация по языку программирования Python : официальный сайт Python [Электронный ресурс] -. URL: <https://www.python.org/doc/>.
6. Szegedy, Ch. Scalable, High-Quality Object Detection [Электронный ресурс] // arXiv: открытый архив научных статей. - URL: <https://arxiv.org/abs/1412.1441>
7. Ryo Takahashi. Data Augmentation using Random Image Cropping and Patching for Deep CNNs [Электронный ресурс] // arXiv: открытый архив научных статей. - URL: <https://arxiv.org/abs/1811.09030>.
8. Wei Liu. SSD: Single Shot MultiBox Detector [Электронный ресурс] // arXiv: открытый архив научных статей. - URL: <https://arxiv.org/abs/1512.02325>
9. Abu-Mostafa, Y. Learning From Data / Yaser S. Abu-Mostafa, Malik Magdon-Ismael, Hsuan-Tien Lin – AMLBook. – 2012. - Jan. -C. 213.
10. Eddison L. Python Machine Learning: A Technical Approach To Python Machine Learning For Beginners / Leonard Eddison - CreateSpace Independent Publishing Platform. – 2018. - Mar. - C. 292.
11. Geron, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow / Aurelien Geron // O'Reilly Media. – 2017. - Mar. - C. 566.
12. Goodfellow, I. Deep Learning (Adaptive Computation and Machine Learning series) / Ian Goodfellow, Yoshua Bengio, Aaron Courville // The MIT Press, 2016. - Nov. - C. 800.

Волгушева А. И., Мальчева Р. В. Применение сверточных нейронных сетей для распознавания объектов на изображении. Выполнен анализ принципов работы искусственных нейронных сетей, сферы их применения для распознавания объектов на изображении. Выбран и реализован алгоритм на основе сверточных нейронных сетей, способный с довольно высокой точностью обнаруживать на изображении выбранные классы объектов. Использование данного алгоритма для анализа видеопотоков, получаемых с касс самостоятельного обслуживания, может позволить предприятию, такому как зал кафе или ресторан быстрого питания, функционировать без участия человека.

Ключевые слова: разработка, нейронная сеть, алгоритм, объект, анализ.

Volgusheva A. I., Malcheva R. V. Application of convolutional neural networks for object recognition in an image. The analysis of the principles of artificial neural networks, the scope of their application for recognizing objects in the image is carried out. An algorithm based on convolutional neural networks has been selected and implemented, capable of detecting selected classes of objects in an image with fairly high accuracy. Using this algorithm to analyze video streams received from self-service ticket offices can allow an enterprise, such as a cafe hall or a fast food restaurant, to function without human intervention.

Keywords: development, neural network, algorithm, object, analysis.

Статья поступила в редакцию 15.06.2024
Рекомендована к публикации профессором Зори С. А.