

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра теоретических основ компьютерной безопасности и криптографии

**ОБНАРУЖЕНИЕ ОБЪЕКТОВ НА ИЗОБРАЖЕНИИ
ИСКУССТВЕННЫМИ НЕЙРОННЫМИ СЕТЯМИ**

КУРСОВАЯ РАБОТА

студента 2 курса 231 группы
направления 100501 — Компьютерная безопасность
факультета КНиИТ
Улитина Ивана Владимировича

Научный руководитель
доцент

Слеповичев И. И.

Заведующий кафедрой

Абросимов М. Б.

Саратов 2021

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	4
1 Концепция технологии	6
1.1 Математическая составляющая нейронной сети.....	6
1.2 Сверточная нейронная сеть	8
1.2.1 Двумерная сверточная нейронная сеть	9
1.2.2 Многоканальная версия сверточной нейронной сети	11
1.3 Метод обратного распространения ошибки	12
1.4 Роль глубокого обучения	15
1.5 Компоненты глубокого обучения.....	15
1.6 Устройство модели глубокого обучения	16
2 Применение технологии	18
2.1 Задачи мониторинга	18
2.2 Задачи анализа визуальных данных	19
2.2.1 Восприятие	20
2.2.2 Локализация	20
2.2.3 Планирование	21
2.2.4 Управление.....	21
2.2.5 Итоги.....	21
ЗАКЛЮЧЕНИЕ	22
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	23

ВВЕДЕНИЕ

В современном мире, в эпоху информационных технологий, с каждым годом увеличивается количество задач, которые могут быть решены с помощью компьютера. И чем сложнее компьютер становится, тем более серьезные проблемы становятся решаемыми. Однако были такие задачи, которые легко решались только человеческим разумом, за счет интуиции, а не вычислительной машины, так как лучше всего компьютером разрешались такие задачи, которые имели математическую составляющую, подход к которым осуществлялся с помощью определенного набора аксиом и правил. С момента появления компьютера, совсем недавно начала развиваться наука об искусственном интеллекте, которая открыла компьютерам возможность находить решения для таких проблем, которые раньше могли быть решены только человеком. Вследствие этого сейчас активно начала использоваться технология нейронных сетей, которая позволяет решать простые человеческие проблемы достаточно быстро, упрощая повседневную жизнь людей. Вот несколько примеров задач, в которых используется искусственный интеллект:

1. Определение самого короткого маршрута из точки А в точку Б с учетом трафика в городе;
2. Перевод с одного языка на другой;
3. Автоматическое определение диагноза пациента при наборе симптомов;
4. Парковка машины с помощью искусственного интеллекта;
5. Сведение звука в музыкальных произведениях;
6. Классификация большого объема данных каких-либо форматов по различным нетривиальным или неоднозначным ключам;
7. Определение объектов на изображении;
8. Формирование прогноза погоды на основе некоторой метеорологической базы данных;
9. Предсказание курса ценных бумаг.

В данной работе будет рассматриваться задача обнаружения объектов на изображении посредством искусственных нейронных сетей, её примеры в реальной жизни, математическая составляющая её решения и анализ соответствующих этому решению инструментов.

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

Перед анализом математической составляющей нейронной сети, работающей с конкретной задачей (в данном случае — с анализом изображений на предмет обнаружения объектов) стоит ввести ряд терминов и определений, которые являются фундаментом понимания работы нейросетей.

Граф — абстрактное математическое понятие, определяющее объект, который состоит из совокупности вершин и ребёр, которые соединяют вершины.

Сетевая топология — вид графа, вершинами которого являются конечные узлы сети, а ребрами - связи между вершинами, содержащие информацию.

Искусственный интеллект (Artificial Intelligence) — технология создания алгоритмов, лежащих в основе проектирования интеллектуальных машин и программ, способных имитировать деятельность человека.

Нейронная сеть (нейросеть) (Neural Network) — математическая модель, чаще всего имеющая программную интерпретацию, сутью которой является реализация деятельности, похожей на деятельность биологических нейронных сетей. Нейронная сеть используется при создании какого-либо из алгоритмов искусственного интеллекта и состоит из совокупности нейронов, соединенных между собой связями.

Искусственный нейрон — преобразователь одного или нескольких входных элементов в по крайней мере один выходной элемент в некоторый дискретный момент времени с определенным шагом.

Входной, или видимый слой (Input layer) — один нейрон или совокупность нейронов в нейросети, содержащая открытые, входные и неизмененные данные, которые включают в себя доступные для изучения переменные.

Скрытый слой (Hidden layer) — один нейрон или совокупность нейронов в нейросети, которая осуществляет вычисления и генерирует данные, не доступные для анализа и не являющиеся изначальными или конечными.

Выходной слой (Output layer) — один нейрон или совокупность нейронов в нейросети, представляющая собой результат работы нейронной сети.

Признак — каждый отдельный элемент информации, включаемый в представление о каком-либо анализируемом объекте.

Машинное обучение (Machine Learning) — область науки об искусственном интеллекте, которая изучает способы создания алгоритмов, которые могут обучаться (развиваться).

Обучение представлений — использование методов машинного обучения для определения представления.

Автокодировщик — совокупность функции кодирования (которая преобразует входные данные в удобное для решения задачи представление) и функции декодирования, являющейся обратной по смыслу к предыдущей функции.

Фактор вариативности — в контексте машинного обучения это концепция, помогающая получить смысл из данных той характеристики об изучаемом объекте, которая может иметь большое количество различных значений, то есть обладающая высокой вариативностью.

Глубокое обучение (Deep Learning) — частный случай машинного обучения, который представляет из себя методы машинного обучения, основанные на обучении представлений. Осуществляет получение представлений путем их выражения через более простые представления, а формирование последних, в свою очередь, реализуется через ещё более простые представления, и так далее.

Компьютерное зрение (Computer Vision) — область науки об искусственном интеллекте, использующая методы машинного и глубокого обучения для решения задач распознавания, классификации, мониторинга с помощью получения необходимой информации из изображения.

1 Концепция технологии

1.1 Математическая составляющая нейронной сети

Для понимания работы алгоритмов машинного обучения, которые активно применяются при реализации компьютерного зрения, следует иметь представления о нейронных сетях, которые эти алгоритмы используют.

Структура нейронной сети представляет из себя результат деятельности нескольких математических дисциплин, среди которых следует выделить несколько важных для нас:

1. Линейная алгебра — область математики, а именно раздел алгебры, который изучает объекты линейной природы: скаляры, векторы, матрицы, определители, сопряжения, линейные отображения, системы линейных уравнений, линейные и векторные пространства и т.д. Она используется во многих алгоритмах машинного и глубокого обучения.
2. Теория вероятностей — область математики, в которой рассматриваются недостоверные утверждения, то есть события, которые могут произойти с определенным шансом на успех. Она определяет средства, с помощью которых можно количественно описать недостоверность событий, а также предоставляет аксиомы для вывода новых недостоверных суждений. Так как машинное обучение работает с стохастическими и неопределенными событиями, применение теории вероятностей необходимо.
3. Теория графов — область математики, изучающая графы, а также связанные с ним понятия, операции и алгоритмы.

В связи с этим, наиболее популярно представление нейронной сети в виде перцептрона. Перцептрон (или же в некоторых источниках персептрон) — это математическая модель, целью которой является интерпретация деятельности человеческого мозга посредством вышеупомянутых областей математики. Другое определение перцептрона — это ориентированный граф, вершины которого описывают входные, скрытые и выходные слои модели машинного обучения, связанные друг с другом с помощью ребер, каждое из которых обладает собственным весом, которое может быть как статично, так и дифференцируемо в зависимости от определения поведения нейронной сети при решении конкретной задачи (однако стоит отметить, что, несмотря на возможность изображения перцептрона в виде графа, никаких практических преимуществ в этом нет, так как большая часть реализаций перцептрона не подразумевает использование

алгоритмов теории графов).

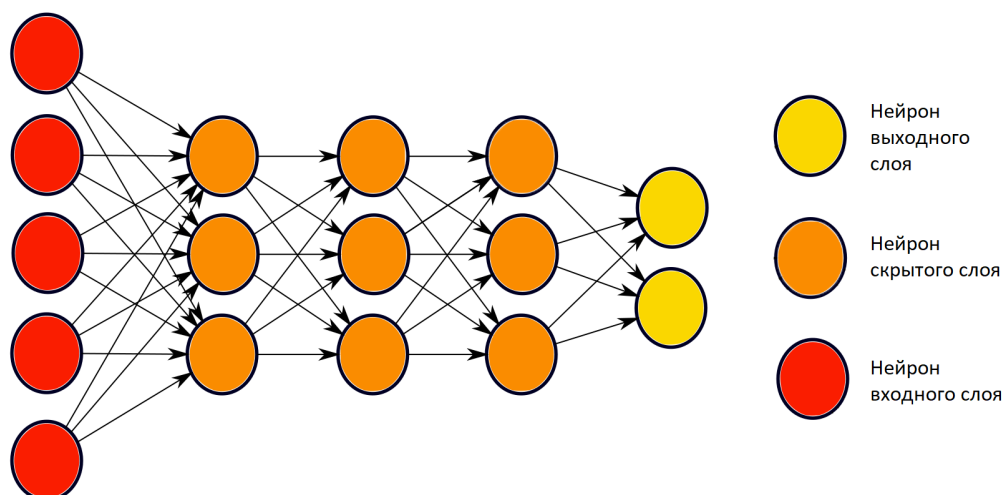


Рисунок 1 – Простая схема многослойного перцептрона

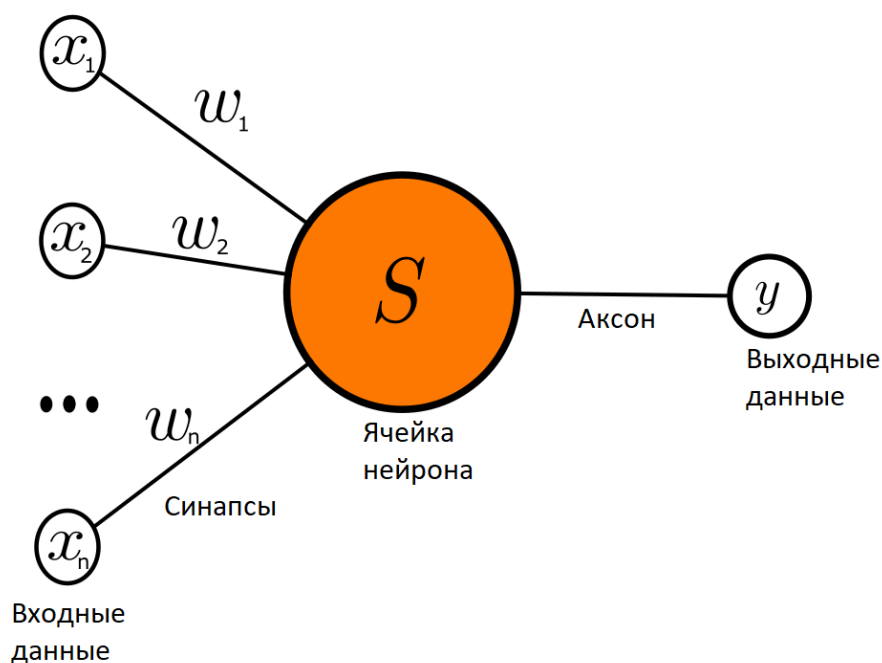


Рисунок 2 – Устройство нейрона

На рис. 1 изображен многослойный перцептрон, состоящий из искусственных нейронов, соединенных друг с другом синапсом (дугой). На рис. 2 представлен нейрон [1], соединяемый с входными данными (это могут быть как исходные данные, так и выходные данные из других нейронов) с помощью синапсов. Каждый синапс имеет собственный вес, определяющий влияние входных данных

нейрона на его состояние, а также на значение, которое будет сформировано нейроном как выходные данные, возвращаемые с помощью аксона. Состояние нейрона, являющееся содержимым ячейки, определяется выражением [2]

$$S = \sum_{i=1}^n x_i w_i, \quad (1)$$

где n — число синапсов нейрона; w_i — вес i -того синапса; x_i — значение i -го входа нейрона.

Выходные данные, передаваемые через аксон, вычисляются по формуле

$$y = f(S), \quad (2)$$

где y — выходные данные; S — выражение состояния нейрона; f — активационная функция, вид которой определяется в зависимости от поставленной задачи [3].

1.2 Сверточная нейронная сеть

В частности, для обнаружения объектов на изображении, может использоваться сверточная нейронная сеть — специальная разновидность нейронной сети, предложенная Яном Лекуном, обрабатывающая данные с сеточной топологией и характеризующая технологию компьютерного зрения. Название этой сети напрямую связано с алгебраической линейной операцией свертки, сутью которой является получение функции из двух других функций. Сверточная сеть предполагает использование операции свертки вместо общей операции умножения на матрицу в по крайней мере одном слое.

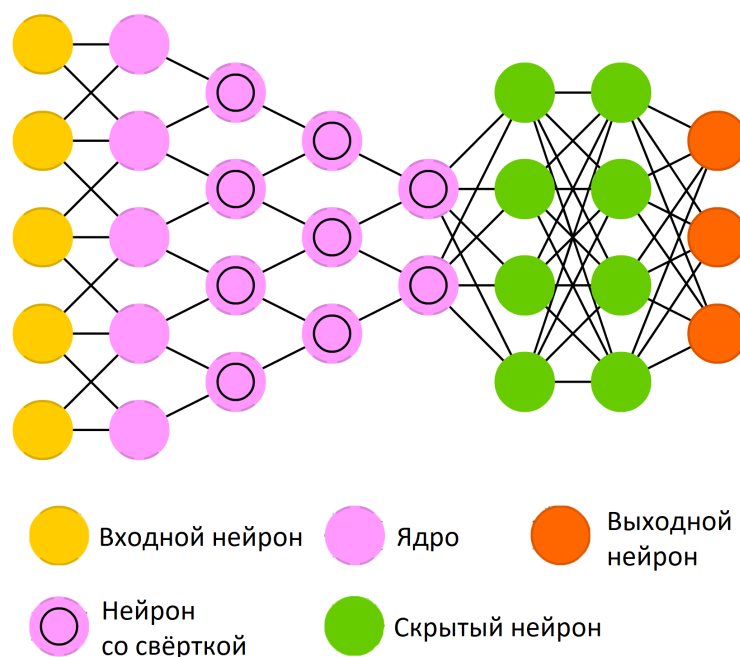


Рисунок 3 – Схема сверточной нейросети

1.2.1 Двумерная сверточная нейронная сеть

В данной нейросети используется операция двумерной свертки. Определим ядро как матрицу весов. Оно перемещается над двумерным изображением, поэлементно совершая операцию умножения с той частью входных данных, над которой оно находится в текущий момент времени, после чего суммирует все полученные значения в один выходной пиксель (см рисунок 4).

Ядро выполняет этот алгоритм каждый раз при перемещении, таким образом преобразуя двумерную матрицу в матрицу признаков. Признаки на выходе являются взвешенными суммами (веса определяются значениями ядра) входных признаков, находящихся в том же месте, что и выходной пиксель на входном слое.

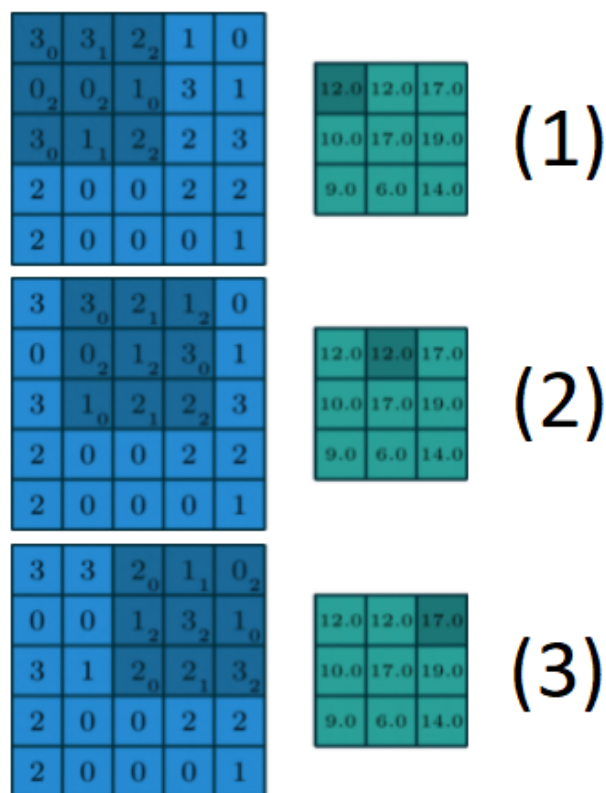


Рисунок 4 – Операция двумерной свертки на примере трех шагов

Входной признак определяется по тому, находится ли он в зоне создающего выходные данные ядра, или нет. Вследствие этого, размер ядра сверточной нейросети определяет количество признаков, которые будут объединены с целью получения нового признака на выходе [4].

В примере на рисунке 4 будет 25 признаков на входе и 9 признаков на выходе. При использовании стандартного слоя необходимо было бы работать с 225 параметрами, а каждый выходной признак был бы взвешенной суммой всех входных признаков. Благодаря свертке можно получить аналогичный результат с 9 параметрами, так как каждый выходной признак является результатом только одного входного.

Таким образом, ранее описанную формальным языком операцию свертки для изображения, определяемого одним каналом, можно определить следующим выражением [5]:

$$y_{i,j} = \sum_{n=1}^S \sum_{m=1}^S w_{n,m} x_{((i-1)+n, (j-1)+m)}, \quad (3)$$

где $w_{i,j}$ — значение элемента ядра свертки на позиции (i, j) ;
 $y_{i,j}$ — значение пикселя выходного изображения на позиции (i, j) ;

$x_{((i-1)+n, (j-1)+m)}$ — значение пикселя входного изображения на позиции $((i-1)+n, (j-1)+m)$;
 S — размерность ядра свертки.

1.2.2 Многоканальная версия сверточной нейронной сети

На практике чаще всего входное изображение состоит из 3 каналов, число которых постепенно увеличивается с глубиной сети.

Количество ядер соответствует количеству входных каналов, причем каждое из ядер является уникальным и определяет свой канал. Совокупность ядер является фильтром, и каждый фильтр в сверточном слое нейросети (сверточный слой — тот, в котором нейроны применяют операцию свертки) создает только один выходной канал по следующему алгоритму: каждое из ядер перемещается по соответствующему ему входному каналу и при этом создает его измененную версию. Веса ядер могут быть различны в зависимости от того, над каким входным каналом необходимо осуществить больше операций (например, фильтр может задавать ядру зеленого канала больший вес, чем другим каналам, для того чтобы лучше обращать внимание на различия в образах, состоящих из зеленого канала).

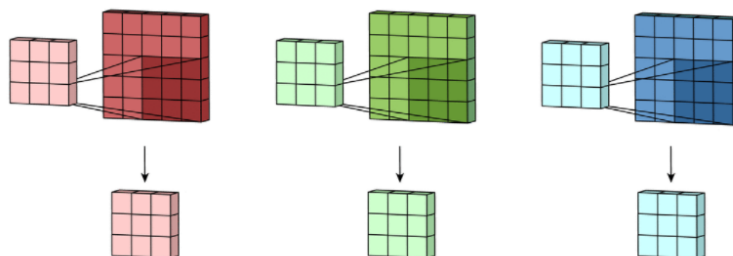


Рисунок 5 – Свертка при трех входных каналах

После получения всех измененных версий входных каналов осуществляется их суммирование для формирования одного канала. Ядро каждого из каналов определяет одну версию конкретного канала, и фильтр создает один общий выходной канал на их основе.

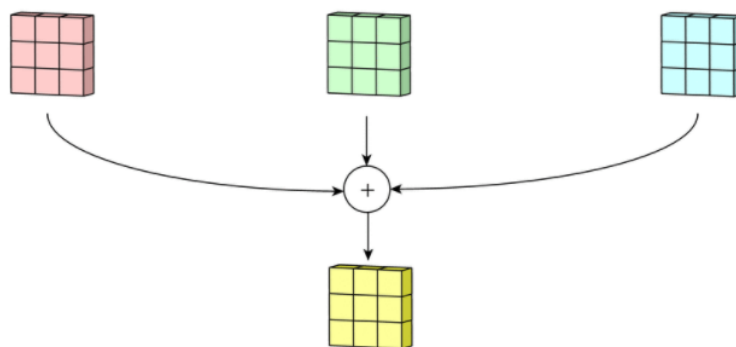


Рисунок 6 – Генерация одного выходного канала

Каждый выходной файл имеет свое смещение. Для получения конечного выходного канала необходимо добавить смещение к каждому выходному каналу:

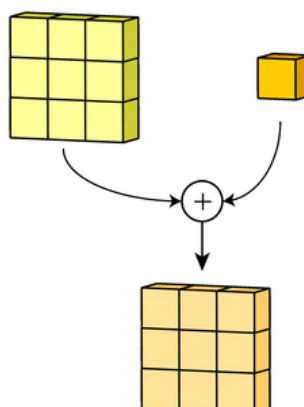


Рисунок 7 – Формирование конечного выходного канала

При любом количестве фильтров результат не меняется: вход обрабатывается каждым фильтром с помощью своего уникального набора ядер и скалярного смещения по представленному ранее алгоритму, и результатом процесса является один выходной канал для одного фильтра. При том, что число выходных каналов соответствует числу фильтров, следующим шагом полученные выходные каналы суммируются, чтобы составить общий выход [4].

1.3 Метод обратного распространения ошибки

Метод обратного распространения ошибки (Backpropagation) является одним из эффективных средств улучшения обучения нейросети и способствующий более продуктивному поиску закономерностей, прогнозированию и качественному анализу. Сети, которые используют этот метод, называются сетями

обратного распространения. Смысл их названия заключается в том, что алгоритм их обучения подразумевает распространение ошибки от выходного слоя к входному, то есть по направлению, противоположному направлению распространения сигнала при естественном функционировании сети [6].

Нейросеть обратного распространения является полносвязной, то есть синапсы между нейронами присутствуют в оба направления сети.

Основной задачей обыкновенного обучения нейросети является определение функции активации, которая в общем случае имеет бесконечное множество решений при ограниченном наборе входных данных. При реализации сети с обратным распространением необходимо ограничить пространство поиска, и для этого ставится задача минимизации целевой функции ошибки нейросети, которая находится с помощью следующей формулы, называемой методом наименьших квадратов:

$$E(w) = \frac{1}{2} \sum_{i=1}^n (y_i - d_i)^2, \quad (4)$$

где y_i — значение i -ого выходного нейрона; d_i — целевое значение i -го выхода; n — количество нейронов в выходном слое.

Основным алгоритмом обучения нейросети является использование градиентного спуска, в котором на каждой итерации (шаге обучения) происходит изменение веса нейрона следующим образом:

$$\Delta w_{ij} = -\eta \cdot \frac{\partial E}{\partial w_{ij}}, \quad (5)$$

где η — параметр, который определяет скорость процесса обучения. В свою очередь,

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial S_j} \cdot \frac{\partial S_j}{\partial w_{ij}}, \quad (6)$$

где y_i — значение i -ого выходного нейрона; S_j — взвешенная сумма входных сигналов, вычисляемая по формуле (1).

Стоит отметить, что в равенстве (6) третий множитель можно определить как

$$\frac{\partial S_j}{\partial w_{ij}} = x_i, \quad (7)$$

где x_i — значение i -ого входного нейрона.

Первый же множитель можно выразить следующим образом:

$$\frac{\partial E}{\partial y_i} = \sum_k \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial S_k} \cdot \frac{\partial S_k}{\partial y_i} = \sum_k \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial S_k} \cdot w_{jk}^{(n)}, \quad (8)$$

где k — число нейронов в n -ом слое.

Стоит ввести вспомогательную переменную, благодаря которой можно определить рекурсивную формулу для определения $(n-1)$ -го слоя:

$$\delta_j^{(n-1)} = \frac{\partial E}{\partial y_j} \cdot \frac{\partial y_j}{\partial S_j}, \quad (9)$$

Если известно состояние n -го слоя, тогда рекурсивная формула будет иметь вид:

$$\delta_j^{(n-1)} = \left[\sum_k \delta_k^{(n)} \cdot w_{jk}^{(n)} \right] \cdot \frac{\partial y_j}{\partial S_j}, \quad (10)$$

Поскольку известен целевой вектор, состоящий из значений, которые должны быть получены сетью в результате работы над данным набором входных значений, можно определить последний слой нейросети:

$$\delta_j^{(N)} = (y_i^{(N)} - d_i) \cdot \frac{\partial y_j}{\partial S_j}, \quad (11)$$

где N — номер последнего слоя сети.

Теперь, имея все необходимые переменные и выражения, можно записать выражение метода градиентного спуска в раскрытом виде:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \delta_j^{(n)} \cdot x_i^n, \quad (12)$$

Учитывая все изложенные ранее формулы и определения [2], получается полный алгоритм обучения нейросети обратного распространения:

1. Подача на вход нейронной сети один из требуемых образов и определение значений выходных нейронов.
2. Определение выходного слоя сети с помощью формулы (11) и расчет изменений весов выходных нейронов по формуле (12).
3. Применить рекурсивную формулу (10) и (12) соответственно, а также рассчитать $\Delta w_{ij}^{(n)}$ для всех остальных слоев сети, то есть $n = N - 1, \dots, 1$.

4. Скорректировать все веса нейросети по формуле:

$$w_{ij}^{(n)}(t) = w_{ij}^{(n)}(t - 1) + \Delta w_{ij}^{(n)}(t). \quad (13)$$

5. Если ошибка существенна, осуществить переход к шагу 1.

1.4 Роль глубокого обучения

Большая часть задач искусственного интеллекта решается в два этапа: корректный подбор признаков, а затем их передача алгоритму машинного обучения. В качестве примера можно взять задачу идентификации объекта по звуку речи. В ней полезным признаком является речевой тракт или же голосовой диапазон. Он позволяет с большой точностью определить, является ли говорящий объект мужчиной, женщиной или ребенком. Но далеко не во всех задачах можно сразу понять, какие признаки стоит выделять: для этого достаточно рассмотреть ситуацию, в которой необходимо написать программу обнаружения автомобилей на фотографиях. Известно, что у автомобилей есть колеса, вследствие чего можно в качестве одного из признаков выбрать наличие колеса. Однако нельзя сказать, что описание колеса на уровне пикселей легкая задача. Колесо характеризуется простой геометрической формой, но его распознавание на изображении нередко бывает осложнено различными факторами, такими как отбрасывание теней, наличие щитка для защиты колеса от грязи, объекты на переднем плане, которые могут закрывать часть колеса, и т. д.

В качестве решений подобной задачи рассматривают использование машинного обучения не только как способа нахождения отображения представления на результат, но и для определения самого представления, то есть, подразумевается использование глубокого обучения. С помощью представлений, которые являются результатом обучения, удастся более точно определить объект на изображении, чем с помощью представлений, созданных вручную. В качестве характерного преимущества подобных реализаций стоит отметить быструю адаптацию ИИ к новым задачам с учетом минимального вмешательства человека в изменение структуры конкретной нейронной сети [7].

1.5 Компоненты глубокого обучения

Важную роль в алгоритме обучения представлений является понятие автокодировщика. Обучение автокодировщиков устроено таким образом, что при

каждой последующей итерации, в ходе которой происходит кодирование и декодирование некоторой информации, каждое новое представление характеризуется всё большим количеством полезных свойств при наименьшей потере обрабатываемой информации.

Сутью моделирования алгоритма обучения признаков или же самого признака является получение различных факторов вариативности. Например, в случае анализа изображения машины такие абстракции как её положение, цвет, яркость солнца и его высота над горизонтом являются факторами вариативности, а в случае анализа изображения лица человека факторами вариативности будут являться его цвет, положение и цвет глаз, их расстояние друг от друга, положение и форма губ, и так далее. Однако большое количество факторов вариативности, получаемых в ходе проектирования признаков или иным способом, следует отбросить в силу их влияния на все данные, доступные анализу в процессе обучения нейронной сети (например, форма губ человека при изучении изображения лица зависит от угла зрения и может быть распознана алгоритмом машинного обучения некорректно) [8].

1.6 Устройство модели глубокого обучения

Определение объекта на изображении, представленном в виде набора значений пикселей, трудно решается при использовании функции отображения множества пикселей в распознаваемый объект. Использование в рассматриваемом случае глубокого обучения на порядок упрощает задачу за счёт разбиения исходного набора пикселей на некоторое количество более простых вложенных отображений, описываемых отдельным слоем модели для каждого. Входные данные в виде изображения представляют из себя входной слой. После видимого слоя идет совокупность скрытых слоев, представляющих из себя слои математических функций, каждый из которых предназначен для вычисления значения, являющегося специфичным для ожидаемого результата (их количество напрямую определяет глубину обучения, так как каждый из них извлекает из анализируемого изображения некоторые признаки, и чем больше таких слоев, тем более абстрактные признаки будут получены из исходных данных). Модель самостоятельно определяет пользу каждой из абстракций при конструировании связей в наблюдаемых данных. Завершающим глубокое обучение и определяющим (в данном случае) объект на изображении является выходной слой. На рисунке 8 в составе модели глубокого обучения представлены 3 скрытых слоя, а

также слой входных данных и слой выходных данных. При анализе изображения, в силу доступности информации об исходных пикселях (определяемых входным слоем), первый скрытый слой определяет границы совокупностей пикселей с помощью сравнения последних по признаку яркости соседних пикселей. Имея характеристики границ множеств пикселей, полученные в результате действия первого скрытого слоя, второй определяет углы и контуры, представляемые в виде набора границ. На основании этих данных третий скрытый слой осуществляет распознавание частей конкретных объектов на изображении, представленных в виде совокупностей углов и контуров отдельного вида. А последний, выходной слой определяет объекты на изображении по информации о частях этих объектов [7].

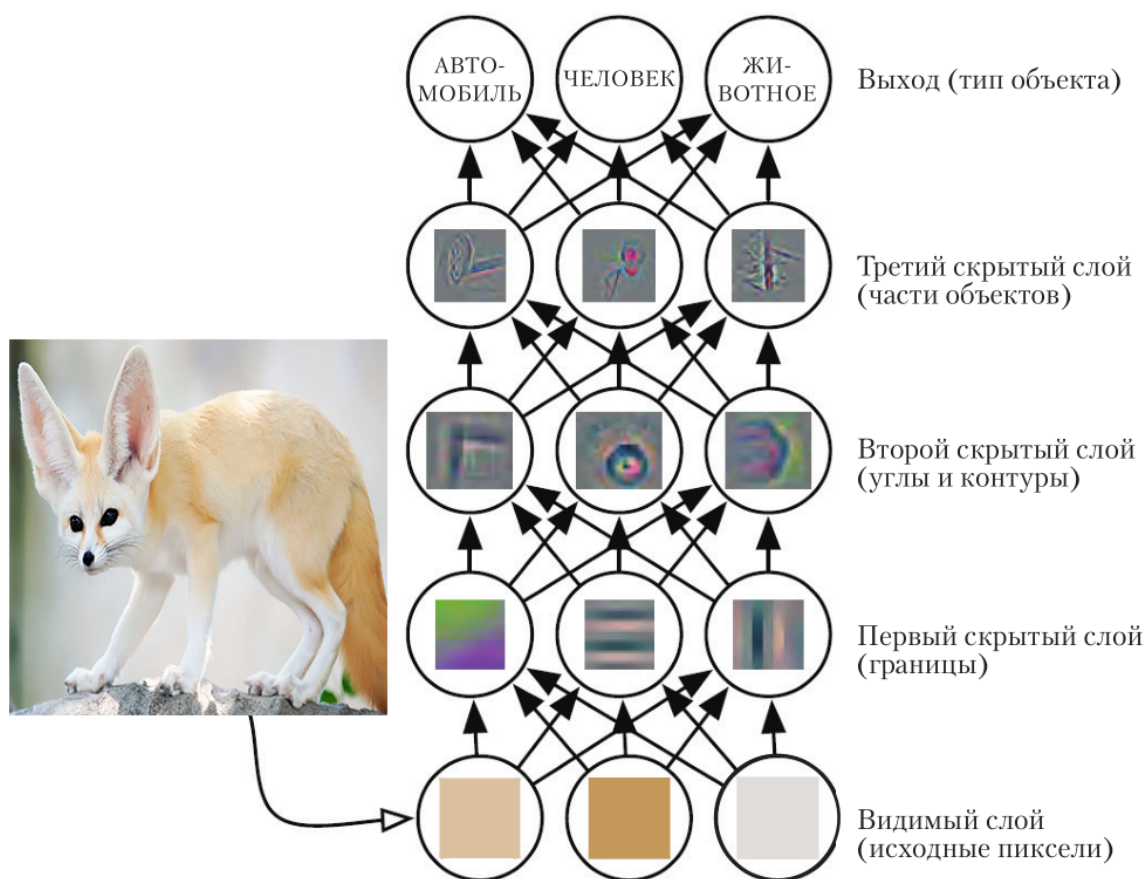


Рисунок 8 – Пример модели глубокого обучения — определение объекта на изображении

2 Применение технологии

Распознавание визуальных образов является одной из самых важных деталей большинства систем управления и анализа информации, а также автоматизированных систем. Задачи, которые связаны с определением предметов, описываемых конечным перечнем особых свойств и признаков, имеют место в некоторых отраслях деятельности машинного обучения, такие как робототехника, мониторинг, анализ визуальных данных, исследования искусственного интеллекта и т.д. Обработка алгоритмами и классификация изображений активно используются в кибербезопасности, в системах контроля и управления доступом, в системах видеонаблюдения, виртуальной и дополненной реальности и информационных поисковых системах. В настоящий момент в производстве активно применяются алгоритмы распознавания рукописного текста, номеров транспортных средств, отпечатков пальцев, человеческих лиц или иных средств аутентификации человека, имеющие огромный спрос при реализации интерфейсов программ, систем безопасности и иных прикладных целей.

2.1 Задачи мониторинга

В качестве примера применения искусственных нейронных сетей, выполняющих обнаружение объектов на изображении в целях мониторинга, можно привести задачу определения разнообразных видов и моделей самолётов, которые находятся на территории аэропорта. Результат работы над конкретной проблемой будет представлять из себя информацию, используемую в различных задачах мониторинга взлётно-посадочных полос (например, определение количества самолетов конкретного вида).

Для выполнения этого задания была использована сверточная нейронная сеть, состоящая из пар слоев — слоев подвыборки и слоев свертки, каждый из которых в свою очередь состоит из карт признаков. Все карты признаков фильтруют изображение, обнаруживая какой-то один определенный, специфичный для конкретной карты, признак [5]. Количество слоев в рассматриваемой нейросети составило 13. Обучение происходило на выборке, определяющейся четырьмя классами объектов: дороги, растительность, городские сооружения и самолеты. Общее количество образцов изображений для обучающей выборки составило приблизительно 5000.

В связи с тем, что анализ изображения происходит с перекрытием (то есть

каждый пиксель изображения анализируется совместно со своими соседними пикселями), повышается способность поиска нужного объекта. Однако при этом появляется возможность многократного вхождения одного и того же объекта сразу в несколько фрагментов. Такой результат работы нейронной сети сложен для обработки, поэтому необходимо применение алгоритма слияния прямоугольников, содержащих объекты.

Нейронная сеть после обработки изображения на выходе совместно с координатными прямоугольниками дает соответствующую каждому из прямоугольников вероятность. Этой вероятностью определяется возможность нахождения координатного прямоугольника над искомым объектом, она определяется числом с плавающей точкой в интервале от 0 до 1. Используемый алгоритм принимает на вход пару, состоящую из координатного прямоугольника и вероятности. Алгоритм слияния состоит из нескольких этапов: создание подсписков из основного списка координатных прямоугольников, фильтрация координатных прямоугольников по вероятности и создание общего координатного прямоугольника [9].

Таким образом, в качестве результата деятельности нейронной сети совместно с алгоритмом слияния удалось решить задачу обнаружения самолета на изображении в целях мониторинга.

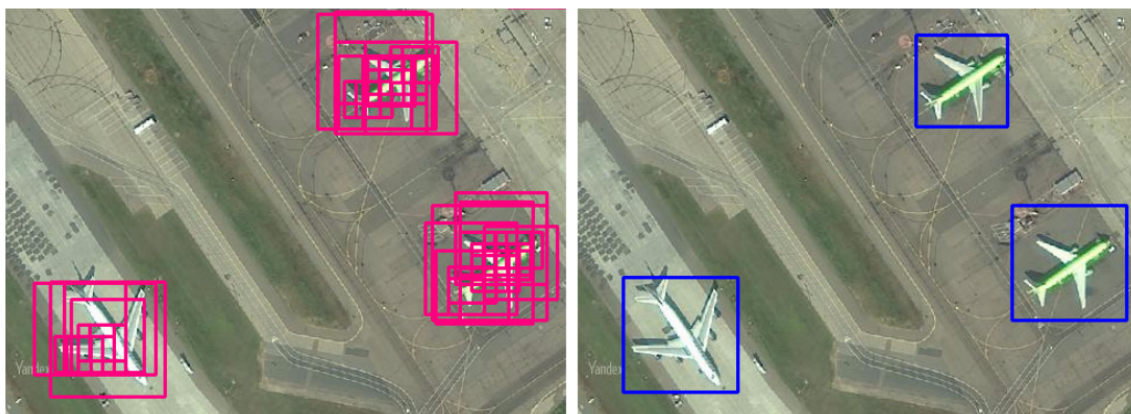


Рисунок 9 – Выделенные объекты интереса до (слева) и после (справа) применения алгоритма слияния

2.2 Задачи анализа визуальных данных

Одним из примеров решения задачи анализа визуальных данных методом обнаружения объекта на изображении с помощью нейросети является развивающаяся технология беспилотного управления автомобилем. [10]

Реализация беспилотных автомобилей состоит из 4 основных пунктов, в каждом

из которых глубокое обучение применяется определенным образом:

1. Восприятие (Perception), то есть целью является анализ окружающей среды и обнаружение различных предметов вокруг точки наблюдения;
2. Локализация (Localization), означающая определение расположения точки наблюдения относительно мира с точностью до 1-3 сантиметров;
3. Планирование (Planning), при котором, используя восприятие и локализацию, вычисляется траектория от точки А до точки В;
4. Управление (Control), сутью которого является использование полученной в предыдущем пункте траектории для создания угла поворота и определения значения ускорения.

2.2.1 Восприятие

Для того, чтобы изучать окружающую среду, необходимо считывать информацию о ней. Это осуществляется с помощью трех видов сенсоров:

1. Камера;
2. Лидар (LiDAR, Light Detection and Ranging);
3. Радар (RADAR, Radio Detection and Ranging).

После получения информации, чтобы удостовериться в её надежности, необходимо совершить слияние данных с сенсоров (Sensor Fusion) и проверить, определяют ли собранные материалы в области наблюдения одни и те же объекты. Слияние происходит только между сведениями, полученными двумя разными путями, например с радара и лидара.

2.2.2 Локализация

В зависимости от выбора алгоритма, различают несколько способов определения местоположения наблюдателя:

1. С помощью знания карты местности и исходного положения до начала движения;
2. Только с помощью знания карты местности;
3. При отсутствии знания карты местности и исходного положения;

Первые две ситуации используют алгоритм обнаружения ориентира (Landmark Detection), так как метод состоит в обнаружении известных объектов и того, что изображено на карте. Идея последней ситуации состоит в том, чтобы использовать датчики, такие как камеры или стереокамеры, для воссоздания окружающей среды и, следовательно, карты.

ЗАКЛЮЧЕНИЕ

В данной работе были рассмотрены следующие темы:

1. Основополагающая теоретическая информация об устройстве нейронной сети и сверточной нейронной сети в частности;
2. Алгоритм свертки для одного или нескольких каналов изображения;
3. Метод обратного распространения ошибки вместе с выводом его пошагового алгоритма со всеми нужными для этого формулами;
4. Важность роли глубокого обучения как средства, позволяющего использовать нейронную сеть для обнаружения объекта на изображении, а также его логика и компоненты.

Также были приведены примеры задач, в которых использование технологии обнаружения объекта на изображении является лучшим решением, в частности, были рассмотрены задачи мониторинга и анализа визуальных данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Короткий С., "Нейронные сети: Основные положения", [Электронный ресурс] : [статья] / URL: http://www.shestopaloff.ca/kyriako/Russian/Artificial_Intelligence/Some_publications/Korotky_Neuron_network_Lectures.pdf, дата обращения 27.04.2021, Яз. рус.
- 2 Короткий С., "Нейронные сети: Алгоритм обратного распространения", [Электронный ресурс] : [статья] / URL: <http://masters.donntu.org/2009/fvti/trubarov/library/article2.htm>, дата обращения 27.04.2021, Яз. рус.
- 3 Dirk Emma Baestaens, Willem Max Van Den Bergh, Douglas Wood, "Neural Network Solution for Trading in Financial Markets", Pitman publishing, 1994 г., Яз. англ.
- 4 Исаков С., "Как работает сверточная нейронная сеть: архитектура, примеры, особенности", [Электронный ресурс] : [статья] / URL: <https://neurohive.io/ru/osnovy-data-science/glubokaya-svertochnaja-nejronnaja-set/>, дата обращения 27.04.2021, Яз. рус.
- 5 Дорогой Я., "Архитектура обобщенных сверточных нейронных сетей", [Электронный ресурс] : [статья] / URL: http://www.it-visnyk.kpi.ua/wp-content/uploads/2012/08/54_36.pdf, дата обращения 27.04.2021, Яз. рус.
- 6 Стариков А., "Нейронные сети — математический аппарат", [Электронный ресурс] : [сайт] / URL: <https://basegroup.ru/community/articles/math>, дата обращения 27.04.2021, Яз. рус.
- 7 Гудфеллоу Я., Бенджио И., Курвилль А., "Глубокое обучение", г. Москва, Издательство ДМК, 2018 г., Яз. рус.
- 8 Sutskever I., Martens J., Dahl G., and Hinton G., "On the importance of initialization and momentum in deep learning.", ICML, 2013 г., Яз. англ.
- 9 Смирнов А. В., Иванов Е. С., "Использование механизма сверточных нейронных сетей для поиска объектов на аэрофотоснимках", [Электронный ресурс] : [статья] / URL: http://psta.psiras.ru/read/psta2017_4_85-99.pdf, дата обращения 10.04.2021, Яз. рус.

- 10 Cohen J., "Deep Learning in Self-Driving Cars", [Электронный ресурс] : [статья] / URL: <https://becominghuman.ai/deep-learning-algorithms-in-self-driving-cars-14b13a895068>, дата обращения 03.05.2021, Яз. англ.