

Школа Инженерная школа ядерных технологий
 Направление подготовки 01.03.02 «Прикладная математика и информатика»
 Отделение школы (НОЦ) Отделение экспериментальной физики

БАКАЛАВРСКАЯ РАБОТА

Тема работы Разработка веб-сервиса по распознаванию 3D-объектов на основе свёрточных нейронных сетей
--

УДК 004.774:004.925.84:004.7.032.26

Студент

Группа	ФИО	Подпись	Дата
0B51	Султонов Ансорхон Азизхонович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ ИЯТШ	Семенов Михаил Евгеньевич	к.ф.-м.н., доцент		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Конотопский Владимир Юрьевич	к.э.н. доцент		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД ШБИП	Исаева Елизавета Сергеевна			

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ ИЯТШ	Крицкий Олег Леонидович	к.ф.-м.н., доцент		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСООБЪЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
0B51	Султонову Ансорхону Азизхоновичу

Школа	ИЯТШ	Отделение школы (НОЦ)	ОЭФ
Уровень образования	Бакалавр	Направление/специальность	01.03.02 Прикладная математика и информатика

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	<ul style="list-style-type: none"> – Принять по действующим ценам – Ставки НДС(20%) и социального найма
2. <i>Нормы и нормативы расходования ресурсов</i>	
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	
2. <i>Планирование и формирование бюджета научных исследований</i>	– Выполнить
3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	– Выполнить

Перечень графического материала (с точным указанием обязательных чертежей):

1. <i>Линейный график работ</i>

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Конотопский Владимир Юрьевич	к.э.н. доцент		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0B51	Султонов Ансорхон Азизхонович		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
0B51	Султонову Ансорхону Азизхоновичу

Школа	ИЯТШ	Отделение (НОЦ)	ОЭФ
Уровень образования	Бакалавр	Направление/специальность	01.03.02 Прикладная математика и информатика

Тема ВКР:

Разработка веб-сервиса по распознаванию 3D – объектов на основе сверточных нейронных сетей	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объектом исследования являются изображения и видео, содержащие группы товаров либо товаров в отдельности. Область применения: Проект можно взаимодействовать с интернет-магазинов, для быстрого нахождения товаров внутри этой сети.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности: <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<ul style="list-style-type: none"> – ГОСТ Р 22.0.02-2016 «Безопасность в чрезвычайных ситуациях» – СНиП 23-05-95 «Естественное и искусственное освещение» – ГОСТ 12.2.032-78 «Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования» – СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы»
2. Производственная безопасность: <ul style="list-style-type: none"> 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия 	<ul style="list-style-type: none"> – Недостаточная освещенность рабочей зоны; – Повышенный уровень статического электричества; – Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека.

3. Экологическая безопасность:	В ходе данной работы были использованы следующие ресурсы: – электроэнергия для работы компьютера; – люминесцентные лампы. При написании ВКР радиационного заражения не произошло и чрезвычайные ситуации не наблюдались.
4. Безопасность в чрезвычайных ситуациях:	– Сильные морозы, диверсия и пожар – Потери при авариях на электросетях – Наиболее типичная ЧС: пожар

Дата выдачи задания для раздела по линейному графику	
--	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ст. преподаватель	Исаева Елизавета Сергеевна			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0B51	Султонов Ансорхон Азизхонович		

РЕФЕРАТ

Выпускная квалификационная работа содержит 76 страниц, 38 рисунков, 9 таблиц, 11 формул, 34 источника, 2 приложения, 2 листинга.

Ключевые слова: компьютерное зрение, распознавание 3D-объектов, классификация изображений, фильтрация, свёрточная нейронная сеть.

Цель работы – разработка компьютерного веб-сервиса для поиска и классификации объектов на графических изображениях (фото-, видеоданные). Для увеличения точности обнаружения объектов мы использовали библиотеки OpenCV, NumPY, Matplotlib для предварительной обработки изображений.

Построенная и обученная сверточная нейронная сеть обеспечивает удовлетворительную точность распознавания объектов при фронтальном расположении и частичную устойчивость при распознавании объектов при их пространственным расположением под углом.

Результаты данной работы по могут быть использованы в коммерческих и личных целях (управление транспортными средствами, поиск товаров по фотографии в интернет-магазине).

Оглавление

Введение.....	8
1. Обзор методов по распознавание объектов, компьютерного зрения и машинного обучения.	10
1.1 Введение в компьютерное зрение	10
1.1.1 Принцип работы компьютерного зрения.....	12
1.2 Обзор методов компьютерного зрения для задачи распознавания объектов	15
1.2.1 Преобразование уровня яркости.....	16
1.2.2 Детектор углов Харриса.....	16
1.2.3 Фильтрации контуров.....	17
1.3 Сверточные нейронные сети	22
1.3.1 Распознавание объектов с помощью нейронной сети	23
1.3.2 Свёрточный слой нейронной сети	25
1.3.3 Подвыборочный слой	27
1.3.4 Полносвязный слой	28
1.4 Обзор программных инструментов для создания веб-сервиса	29
2.1 Сравнение характеристик систем распознавания объектов	32
1.5 Постановка задачи	33
2. Результаты проведенного исследования	35
2.2 Алгоритмы для распознавания объектов.....	36
2.3 Распознавание 3D – объектов.....	38
3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	42
3.1. Организация и планирование работ	42
3.1.1. Продолжительность этапов работ.....	43
3.2. Расчет смены затрат на выполнение проекта	47
3.2.1. Расчет затрат на материалы	47
3.2.2. Расчет заработной платы.....	48
3.2.3. Расчет затрат на социальный налог	49
3.2.4. Расчет затрат на электроэнергию	49
3.2.5. Расчет амортизационных расходов	51
3.2.6. Расчет прочих расходов	53
3.2.7. Расчет общей себестоимости разработки.....	53
3.2.8. Расчет прибыли	53
3.2.9. Расчет НДС.....	54
3.2.10. Цена разработки НИР	54
3.3. Оценка экономической эффективности проекта	54

4. Социальная ответственность.....	55
4.1 Правовые и организационные вопросы обеспечения безопасности	56
4.1 Производственная безопасность	59
4.2 Анализ опасных и вредных производственных факторов и обоснование мероприятий по их устранению	60
4.3.1 Недостаточная освещенность рабочего места	60
4.3.2 Повышенный уровень статического электричества.....	61
4.3.3 Повышенное значение напряжения в электрической цепи.....	61
4.3.4 Повышенная или пониженная температура воздуха рабочей зоны	63
4.3 Экологическая безопасность.....	64
4.1 Безопасность в чрезвычайных ситуациях.....	64
Выводы и рекомендации по разделу	66
Заключение.....	68
Список литературы	73

Введение

Сегодня одними из важнейших областей исследований и разработок современной прикладной математики и кибернетики являются – компьютерное зрение, распознавание образов, машинное обучение, а также прогнозирование и анализ данных. Ускоряющиеся темпы развития технологий информационного общества, развитие концепций «умный дом» и «умный город», развитие интернета и системы искусственного интеллекта определяют особое место этим областям в современном мире. Во многих прикладных задачах программирования используются методы сбора данных, кластеризации и классификации, статистического вывода. В повседневную жизнь активно внедряются технологии распознавания образов. Ранее считавшиеся сложными задачи распознавания, сегодня решаются с помощью обычных мобильных устройств и смартфонов.

Распространённые методами и подходами к решению задач детектирования, распознавания и классификации являются:

- *Сравнение с образцом* – классификация по ближайшему среднему, по расстоянию до ближайшего соседа. Также в группу сравнения с образцом можно отнести структурные методы распознавания.

- *Сопоставление с шаблоном* – метод распознавания, в котором используется небольшое изображение или шаблон для поиска совпадающих областей в увеличенном изображении.

- *Нейронные сети* – класс методов глубокого обучения, который используется для автоматического изучения свойств объекта и его дальнейшей идентификации. Отличительной особенностью этих методов от других является способность обучаться.

Цель данного проекта – создание компьютерного веб-сервиса для поиска и классификации объектов на графических изображениях (фото-, видеоданные).

Для достижения поставленной цель в работы были поставлены и решены следующие задачи:

1. Рассмотреть существующие алгоритмы для детектирования и классификации объектов.
2. Выбрать языки и среды программирования для реализации выбранных алгоритмов.
3. Программно реализовать сервис по распознаванию объектов.
4. Обучить классификатор с использованием разработанных моделей и провести его тестирование.

1. Обзор методов по распознавание объектов, компьютерного зрения и машинного обучения.

1.1 Введение в компьютерное зрение

Компьютерное зрение (computervision, машинное зрение, техническое зрение) – это технология, которая в зависимости от поставленной задачи, может находить, отслеживать, классифицировать и идентифицировать объекты, извлекая и анализируя полученную информацию из изображений или видео. Это направление возникло в рамках искусственного интеллекта. Основной задачей является распознавание образов, поэтому для полной и правильной интерпретации того, что изображено, нужно иметь необходимую информацию как массив пикселей, извлеченных из изображения. Каждое изображение состоит из набора пикселей. Для компьютера нет более тонкой детализации кроме пикселя.

Под компьютерным зрением, также понимается автоматическое извлечение информации из изображений. В роли информации может выступать 3D-модели, положение камеры, обнаружение и распознавание объектов, группировка изображений и поиск изображений по содержанию. Применение компьютерного зрения приобретает довольно большую значимость в различных сферах нашей жизни. С помощью этой технологии реализована умная системы видеонаблюдения, которая обрабатывает входные видеоданные и, в соответствие обученного алгоритма, принимает необходимые решения.

К решению задач детектирования и классификации применяются различные подходы: статистические, специально разработанные теории ключевых точек, применение алгоритмами классификации изображений по содержанию, а также востребованный подход –машинное обучение.

Наиболее распространенные задачи компьютерного зрения [10]:

– *Распознавание*– одна из базовых и первостепенных задач в обработке изображений, компьютерном и машинном зрении. Оно помогает классифицировать и идентифицировать объекты, характеризующиеся определенным набором свойств и признаков.

– *Восстановление изображений* – это удаление шума с использованием разных методов, например, размытие с помощью фильтров на основе машинного обучения (шум датчика, размытость движущегося объекта и т. д.).

– *Анализ движения* – задача использует компьютерное зрение для оценки скорости движения объектов в видео. Также применяется для оценки движений, в которых последовательность видеоданных обрабатывается для нахождения скорости каждой точки изображения или 3D сцены.

– *Восстановление или реконструкция сцены* – помогает воссоздать трехмерную модель изображения или сцены, вводимой с помощью изображений или видео. Чаще всего, моделью служит набор точек трехмерного пространства.

– *Обработка и анализ изображения*– задача сосредоточена на преобразовании одного 2D-изображения в другое. Реализуется с помощью пиксельных операций, таких как повышение контрастности или поворот изображения.

– *Высокоуровневая обработка* – представляет небольшой набор данных. Она использует различные методы для извлечения информации из сигналов в целом, например, набор точек или участок изображения, в котором предположительно находится определенный объект, либо интересующая часть данных.

1.1.1 Принцип работы компьютерного зрения

Компьютеры интерпретируют изображения как последовательность пикселей, каждый из которых имеет свой собственный набор значений цвета. Пиксели являются необработанными строительными блоками изображения. Каждое изображение состоит из набора пикселей. Обычно пиксель считается «цветом» или «яркостью» света, который появляется на изображении [2]. Если мы рассматриваем изображение как сетку, то каждый квадрат содержит один пиксель.

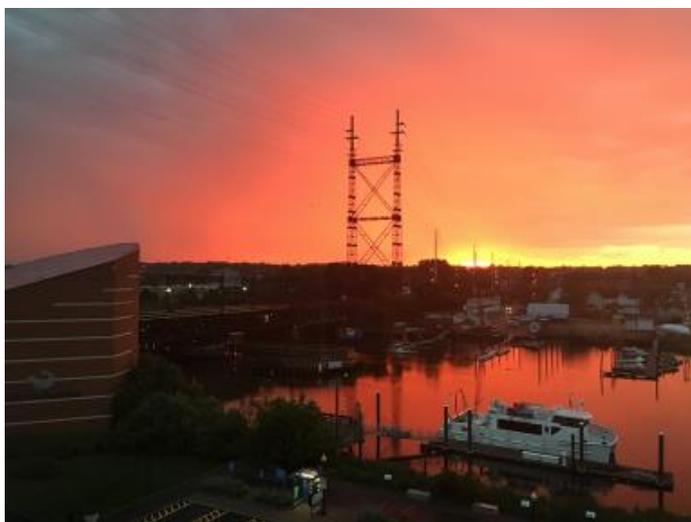


Рисунок 1 – Пример изображения
шириной 1000 пикселей и высотой 750 пикселей

Изображение на рисунке 1 имеет разрешение 1000×750 пикселей, где 1000 это ширина, а 750 – высота. Мы можем представить изображение в виде матрицы. В этом случае наша матрица имеет 1000 столбцов (ширина) и 750 строк (высота) и содержит $1000 \times 750 = 750\,000$ пикселей, которые представлены двумя способами: а) оттенки серого (один канал), б) цвет.

В изображениях с градациями серого цвета каждый пиксель представляет собой скалярное значение от 0 до 255, где ноль соответствует «черному» цвету, а 255 – «белому». Значения между 0 и 255 имеют различные оттенки серого, где значения ближе к 0 темнее, а значения ближе

к 255 светлее. Градиентное изображение в градациях серого (рисунок 2) демонстрирует более темные пиксели с левой стороны, а более светлые с правой. Из рисунка 2, можно понять то, как значения в градациях серого преобразуются в двумерный массив целых чисел[9]:

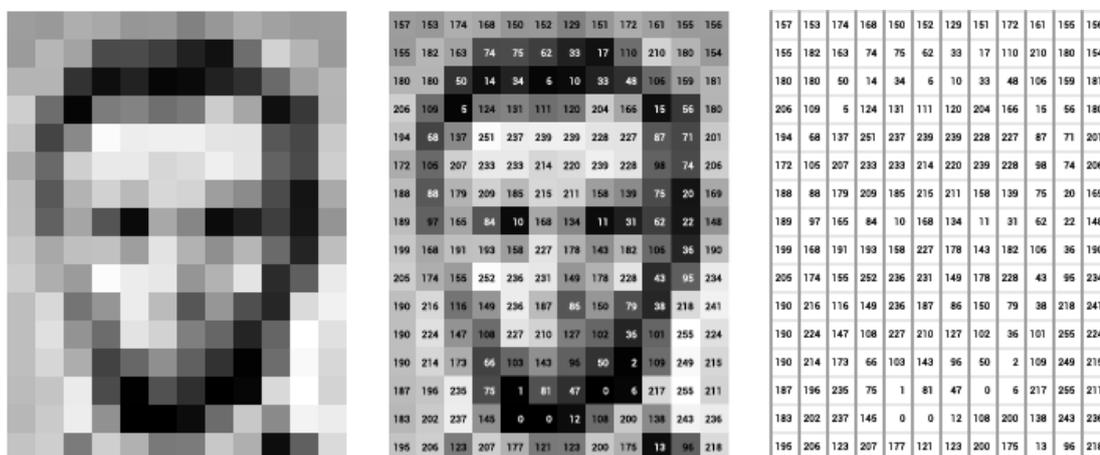


Рисунок 2 – Изображения и их матричные представления

Ряды чисел справа (крайний правый рисунок) – компьютерное представление введенного изображения. В примере изображение имеет 12 столбцов и 16 строк, что означает 192 входных значений для этого изображения.

Ещё необходимо описать как компьютер преобразует для себя цветные изображения в виде матриц. Пиксели в цветовом пространстве RGB больше не являются скалярными значениями, как было в изображениях в градациях серого, на одном канале. Вместо этого пиксели представлены списком из трех значений: одно значение для компонента красного (Red), второе для зеленого (Green) и третье для синего (Blue) [9]. Чтобы определить цвет в цветовой модели RGB, все, что нам нужно сделать, это подсчитать количество красного, зеленого и синего цвета, содержащихся в одном пикселе. Каждый канал Red, Green и Blue может иметь определенные значения в диапазоне $[0, 255]$, всего 256 «оттенков», где 0 означает отсутствие представления, а 255 – это полное представление. Учитывая, что значение пикселя должно быть только в диапазоне $[0, 255]$, мы обычно

используем 8-битные целые числа без знака для представления яркости. Рассмотрим пример создание разных цветов в изображение на рисунке 2а.

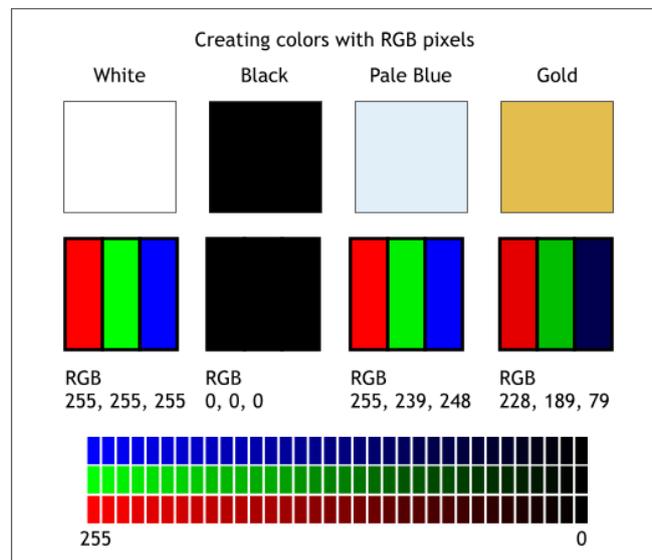


Рисунок 2а – Создание цветов в RGB пикселе

Из этого рисунка 2а, можно понять, что каждый пиксель состоит из трех разных цветов, при изменении их оттенков можно получить абсолютно разные цвета.

Для решения задач детектирования, распознавание и классификации объектов на графическом изображении необходимо предварительно обрабатывать изображение (например, выполнить операцию среднее вычитание или масштабирование). Поскольку типы данных, используемые библиотеками (например, OpenCV), загружают изображения с диска, то их необходимо преобразовывать, прежде чем напрямую применять алгоритмы обучения к изображениям. Учитывая наши три значения Red, Green и Blue, мы можем объединить их в кортеж RGB (красный, зеленый, синий). Этот кортеж представляет данный цвет в цветовом пространстве RGB.

Мы можем преобразовать изображение RGB как состоящее из трех независимых матриц, шириной W и высотой H , по одной для каждого из компонентов RGB, как показано на рисунке 3. Мы можем объединить эти три матрицы для получения многомерного массива с формой $W \times H \times D$, где D -

глубина или количество каналов. Для цветового пространства RGB глубина $D = 3$.



Рисунок 3 – Исходное изображение и его RGB каналы

1.2 Обзор методов компьютерного зрения для задачи распознавания объектов

Для компьютеров задача интерпретация содержимого изображения менее тривиальна, чем просто отображение изображения. Всё, что видит наш компьютер— это большая матрица чисел. Чтобы понять содержание изображения, мы должны применить классификацию изображений, которая является задачей использования алгоритмов компьютерного зрения и машинного обучения [9]. В данной работе были применены различные методы, помимо глубокого обучения, улучшающие компьютерное зрение. Тем не менее, они хорошо работают для более простых задач, но поскольку данные становятся огромными, а задача становится сложной, они не заменяют сверточные нейронные сети.

1.2.1 Преобразование уровня яркости

Считав с помощью NumPy [1] изображение в массив, мы можем применить к нему различные математические операции. Простой пример – преобразование уровня яркости полутонового изображения. Возьмем произвольную функцию f , отображающую интервал $0\dots255$ (или, если угодно, $0\dots1$) в себя, т. е. область значений совпадает с областью определения.

Другим примером преобразования яркости является *выравнивание гистограммы*. Эта операция изменяет гистограмму яркости, так чтобы результирующая гистограмма содержала все возможные значения яркости и при этом примерно в одинаковом количестве. Она часто применяется для нормировки яркости перед последующей обработкой, а также для повышения контрастности. В данном случае для преобразования используется функция распределения (cumulative distribution function, CDF) значений пикселей в изображении. Пример данной операции приведен на рисунке 19.

1.2.2 Детектор углов Харриса

Алгоритм обнаружения углов Харриса (детектором углов Харриса-Стивенса) – один из простейших детекторов углов объектов [2]. Идея заключается в том, чтобы найти особые точки, в окрестности которых имеются границы в нескольких направлениях, это и есть угловые точки.

Определим положительно-полуопределенную симметричную матрицу

$$M_I = M_I(x),$$

где x – точка внутри изображения:

$$M_I = \nabla I \nabla I^T = \begin{bmatrix} I_x \\ I_y \end{bmatrix} \begin{bmatrix} I_x & I_y \end{bmatrix} = \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}. \quad (1)$$

Здесь ∇I – вектор градиента изображения, содержащий производные I_x и I_y (определения производных были даны выше). По построению, M_I имеет ранг 1, а ее собственные значения равны

$$\lambda_1 = |\nabla I|^2, \lambda_2 = 0.$$

Таким образом, у нас имеется по одной матрице для каждой точки изображения.

В зависимости от значений ∇I в области существуют три случая:

- Если λ_1 и λ_2 – большие положительные числа, то в точке x имеется угол.
- Если λ_1 велико, а $\lambda_2 \approx 0$, то существует граница и при усреднении M по области собственные значения изменяются не сильно.
- Если $\lambda_1 \approx \lambda_2 \approx 0$, то в точке нет никаких особенностей.

Рассмотрим метод на примере изображения главного корпуса ТПУ:



а)

б)

Рисунок 4 – а) исходное изображение, б) изображение с обнаруженными углами по алгоритму Харриса

1.2.3 Фильтрации контуров

Контурные изображения очень полезны, когда мы хотим перейти от работы с изображением к работе с объектами на этом изображении. Когда объект

достаточно сложный, но хорошо выделяемый, то зачастую единственным способом работы с ним является выделение его контуров.

Оператор Кэнни является популярным алгоритмом для обнаружения границ и чаще всего используется для выделения контура объектов [18].

Алгоритм Кэнни для обнаружения границ объектов состоит из пяти этапов:

1. Подавление шума;
2. Расчет градиента;
3. Подавление краев изображения;
4. Двойной порог;
5. Отслеживание краев по гистерезису.

Еще одна важная вещь, которую стоит упомянуть, что алгоритм основан на изображениях в градациях серого. Следовательно, предварительным условием является преобразование изображения в оттенки серого цвета перед выполнением вышеупомянутых шагов.

Кратко дадим определения этапов алгоритма Кэнни.

Подавление шума – является один из способов избавиться от шумов на изображении. Для того чтобы сгладить шум применяют размытие по Гауссу. Для этого применяется метод свертки изображений с гауссовым ядром (например, 3x3, 5x5, 7x7 и т. д. пикселей). Размер ядра зависит от ожидаемого эффекта размытия. В основном самое маленькое ядро – это менее заметное пятно. При обработке изображении, ядро (свертка матрицы) является небольшой матрицей. Оно используется для размытия, повышения резкости, тиснения, обнаружения краев и многого другого.

Расчет градиента– определяет интенсивность и направление края путем вычисления градиента изображения с использованием операторов обнаружения края. *Градиент* – это векторная величина, показывающая направление наискорейшего возрастания двумерная функция яркости изображения.

Края соответствуют изменению интенсивности пикселей. Чтобы обнаружить её, проще всего применить фильтры, которые выделяют это изменение интенсивности в обоих направлениях: горизонтальном (x) и вертикальном (y).

Подавление краев изображений—в идеале конечное изображение должно иметь тонкие края. Таким образом, мы должны выполнить не-максимальное подавление, чтобы обнаружить края. Для этого можно применить следующий алгоритм. Обойти через все точки на матрице интенсивности градиента и найти пиксели с максимальным значением в направлениях ребер (рисунок 5).

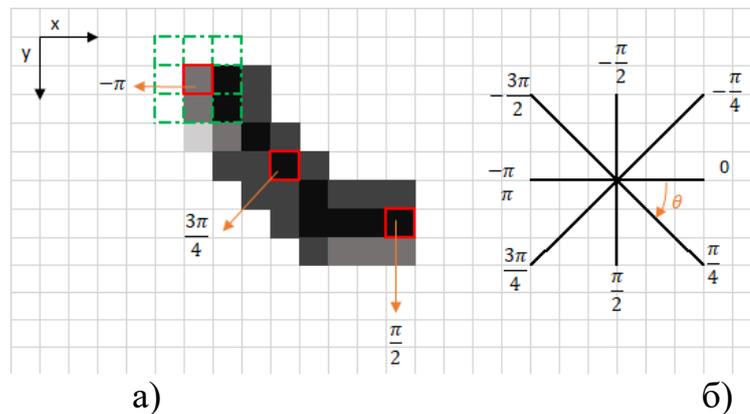


Рисунок 5 – а) часть края объекта в виде пикселей

Красный прямоугольник в верхнем левом углу (рисунок 5а), представляет пиксель интенсивности обрабатываемой матрицы градиентной интенсивности. Соответствующее направление краев обозначено оранжевой стрелкой с углом $-\pi$ радиан.

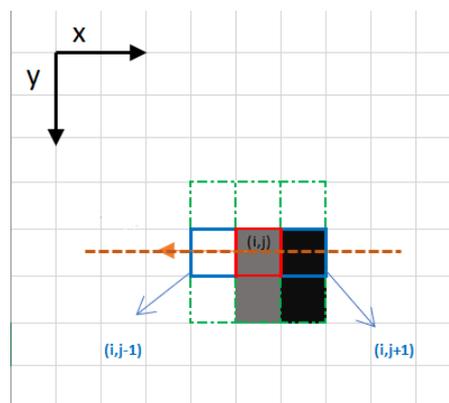


Рисунок 6 – изображение с неполным подавлением краев

Направление края – оранжевая пунктирная линия (горизонтальная линия слева направо, рисунок 6). Цель алгоритма – проверить, являются ли пиксели в одном и том же направлении более или менее интенсивными, чем обрабатываемые [19].

В приведенном выше примере пиксель (i, j) обрабатывается, и пиксели в том же направлении выделяются синим цветом $(i, j-1)$ и $(i, j+1)$. Если один из этих двух пикселей является более интенсивным, чем обрабатываемый, то сохраняется только более интенсивный. Пиксель $(i, j-1)$ кажется более интенсивным, потому что он белый (значение 255). Следовательно, значение интенсивности текущего пикселя (i, j) устанавливается в 0. Если в направлении края отсутствуют пиксели, имеющие более интенсивные значения, то значение текущего пикселя сохраняется [19].

Двойной порог – цель применения данной операции направлена на выявление трех видов пикселей и с помощью них выявлять контуры:

- Сильные пиксели – пиксели с высокой интенсивностью (яркостью).
- Слабые пиксели – это пиксели, которые имеют достаточного значение интенсивности. Эти пиксели нельзя считать сильными, но их значение интенсивности не являются маленькими, чтобы их считать, как пиксели, не имеющие отношение к краям.
- Другие пиксели считаются не имеющие отношение к краям.

Отслеживание краев – результаты порогового значения гистерезис, состоит из преобразования слабых пикселей в сильные, если хотя бы один из пиксель вокруг обрабатываемого края является сильным [19].

Посмотрим полученные нами результаты после применения алгоритма обнаружения краев для разных объектов (рисунки 7-8).

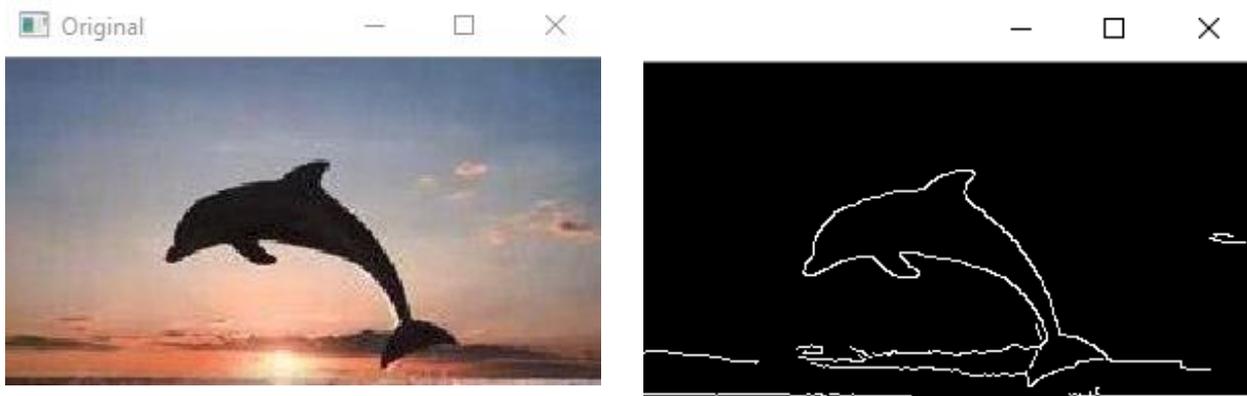


Рисунок 7 – а) исходное цветное изображение,
б) результат обнаружения краев объекта

В результате видно, что данный метод может найти все края объекта (дельфин), удаляя при этом многие края объектов на заднем фоне (облака). Далее рассмотрим более сложный объект и применим тот же самый алгоритм (рисунок 8).

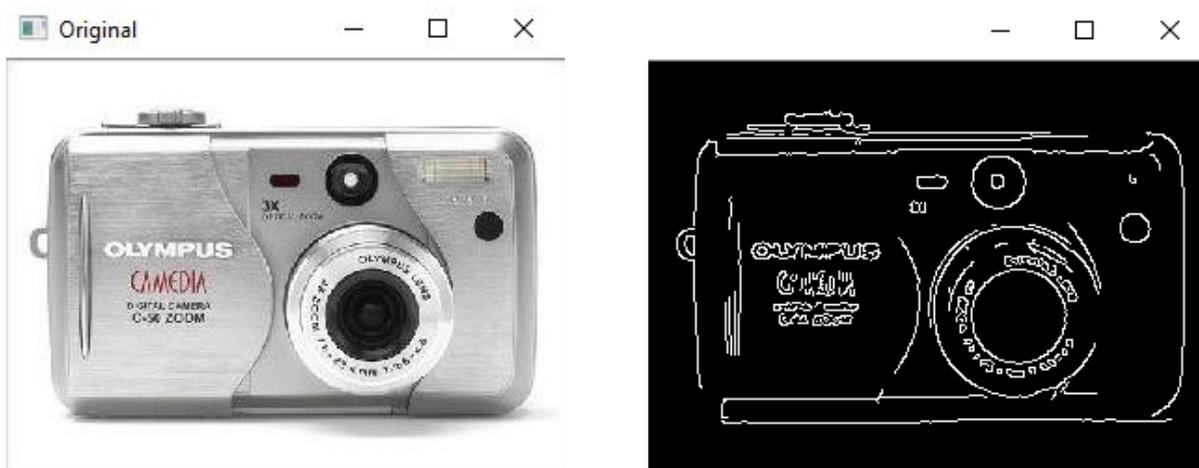


Рисунок 8 – а) исходное цветное изображение,
б) результат обнаружения краев объекта

Из примеров видно, что алгоритм обнаружения краев Кэнни, дает удовлетворительный результат, как и для простых, так и для более сложных объектов.

1.3 Сверточные нейронные сети

Искусственные нейронные сети (нейросети или просто сети) – это класс моделей машинного обучения, в основе которых лежат центральной нервной системы млекопитающих.

Нейронная сеть состоит из нескольких взаимосвязанных различных слоев, таких как *входной* слой, по меньшей мере один *скрытый* слой и *выходной* слой (рисунок 9). Их лучше всего использовать при обнаружении объектов для распознавания образов, краев (вертикальные / горизонтальные), формы, цвета и текстуры. Скрытые слои являются *сверточными* слоями. В данном типе нейронной сети, сверточные слои действует как фильтр, который сначала получает входные данные, преобразует их, используя определенный алгоритм или функцию, и отправляет его на следующий слой. Основные параметры нейронов является входной (синий цвет) и выходной слой(зелёный цвет).

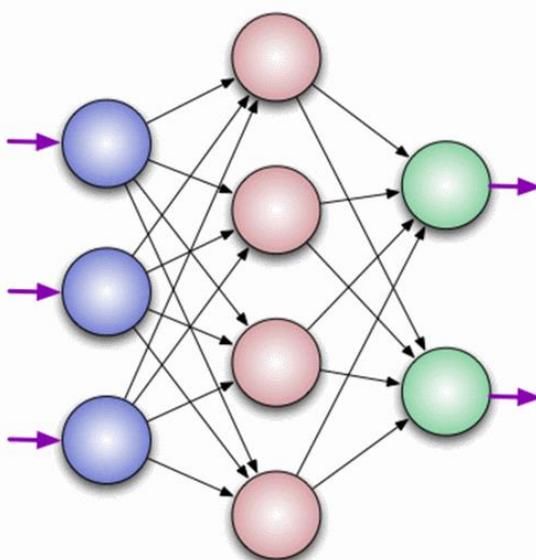


Рисунок 9 –Модель нейронной сети

С большим количеством сверточных слоев, каждый раз, когда новый вход отправляется на следующий сверточный слой, он изменяется по-разному. Например, в сверточном слое фильтр может идентифицировать форму / цвет

в определенной области, последний сверточный слой, может классифицировать объект.

В общем случае сверточная нейронная сеть состоит из большого числа слоев. На последних этапах обычно используется один или несколько полносвязных слоев.

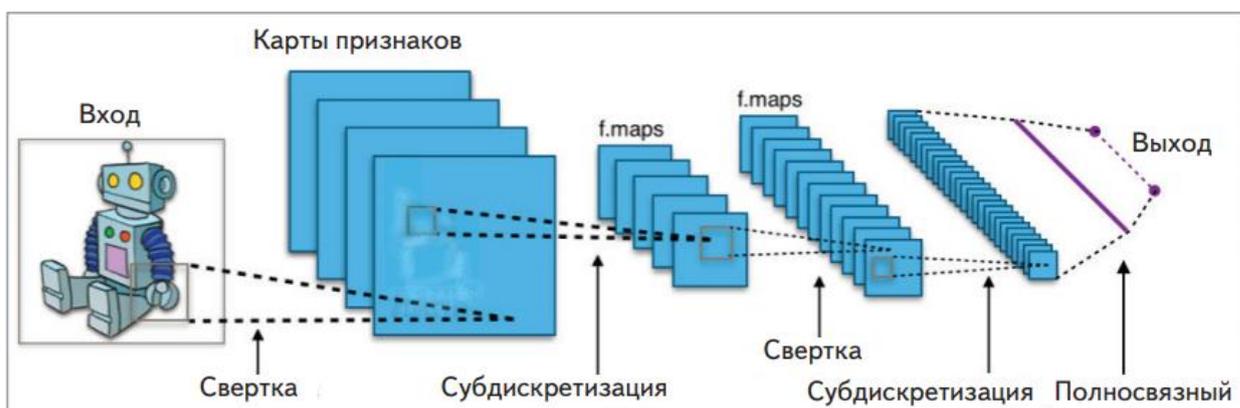


Рисунок 10 – Топология сверточной нейронной сети

Сверточные нейронные сети обеспечивают частичную устойчивость к изменениям масштаба, смещениям, поворотам, смене ракурса и прочим искажениям. Сверточные нейронные сети объединяют три архитектурных идеи, для обеспечения инвариантности к изменению масштаба, повороту сдвигу и пространственным искажениям:

- локальные рецепторные поля (обеспечивают локальную двумерную связность нейронов);
- общие синоптические коэффициенты (обеспечивают детектирование некоторых черт в любом месте изображения [19] и уменьшают общее число весовых коэффициентов);
- иерархическая организация с пространственными подвыборками.

1.3.1 Распознавание объектов с помощью нейронной сети

Для того, чтобы научить нейронную сеть обнаруживать объектов на любом изображении, с приблизительно одинаковой формой и цветом, следует применять разные фильтры (рисунок 11).

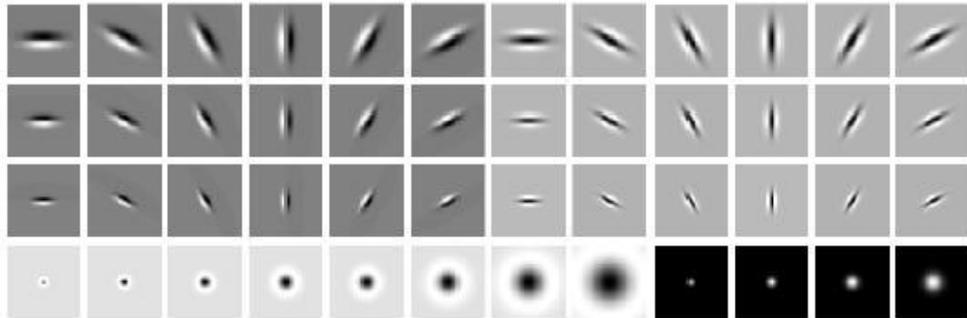


Рисунок 11 – Фильтры для различных фрагментов изображения

С помощью разных фильтров можно выделять различные фрагменты изображения, которые затем получается обнаружить и исследовать в виде отдельных свойств и передавать другими слоями нейрона (рисунок 11).

Для того, чтобы сетям не приходилось отдельно распознавать объекты в разных частях изображения, мы «разделяем» веса, отвечающие за распознавание, между различными фрагментами исходного изображения. Рассмотрим изображение, на котором необходимо не просто выделить объект, а установить количественную точность решения для четырех фрагментов (рисунок 12).

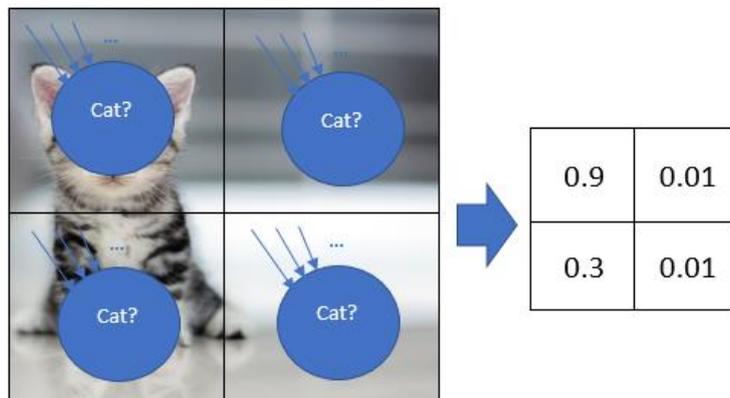


Рисунок 12 – Распознавание объекта и характеристики точности решения

В таблице указаны количественные характеристики решения, которые показывают степень точности распознавания объекта в данном фрагменте изображения. Находим максимальное значение

$$\max\{0.9, 0.3, 0.01, 0.01\}=0.9,$$

которое и будет требуемой характеристикой (рисунок 13).



Рисунок 13 – Получение результата для выявленного объекта

Основная идея алгоритма заключается в следующем:

- Используем разделение весов (weight sharing) для создания «фильтрующего окна», пробегающего по изображению.
- Примененный к изображению фильтр помогает выделить фрагменты, важные для распознавания.
- В то время как в традиционном машинном зрении фильтры конструировали вручную, нейросети позволяют нам сконструировать оптимальные фильтры с помощью обучения.
- Фильтрацию изображения можно совместить с вычислениями нейронной сети.

1.3.2 Свёрточный слой нейронной сети

Свёрточный слой представляет собой набор карт (карты признаков, features maps). Каждая карта имеет сканирующее ядро (сканирующее ядро - представляет из себя фильтр, который скользит по всему изображению и находит заданные признаки в любом его месте). Количество карт определяется требованиями к задаче, если взять большое количество карт, то

повысится качество распознавания, но увеличится вычислительная сложность.

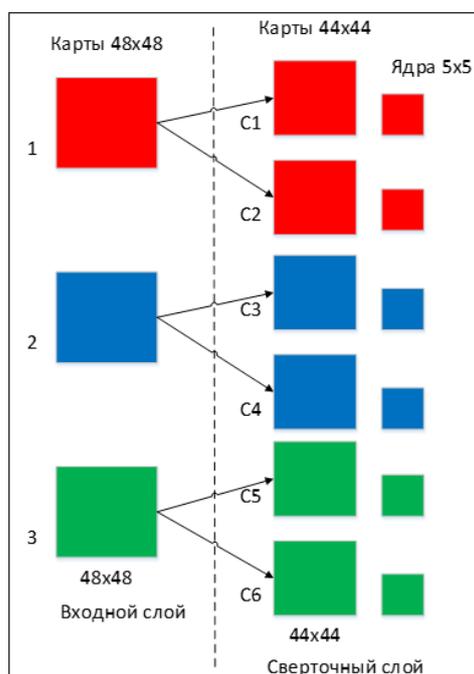


Рисунок 14 – Организация связей между картами сверточного и входного (предыдущего) слоя

Размер у всех карт сверточного слоя одинаковый и вычисляется по формуле:

$$(w, h) = (mW - kW + 1, mH - kH + 1) \quad (2)$$

где (w, h) – вычисляемый размер сверточной карты, mW – ширина предыдущей карты, mH –высота предыдущей карты, kW – ширина ядра, kH – высота ядра.

Ядро скользит по предыдущей карте и производит операцию свертка, которая часто используется для обработки изображений:

$$(f * g)[m, n] = \sum f[m - k, n - l] \cdot g[k, l] \quad (3)$$

где f – исходная матрица изображения, g – ядро свертки.

Происходит следующие: окном размера ядра g проходим с заданным шагом (обычно 1) все изображение f , на каждом шаге поэлементно умножаем содержимое окна на ядро g , результат суммируется и записывается в матрицу результата, как на рисунке 15.

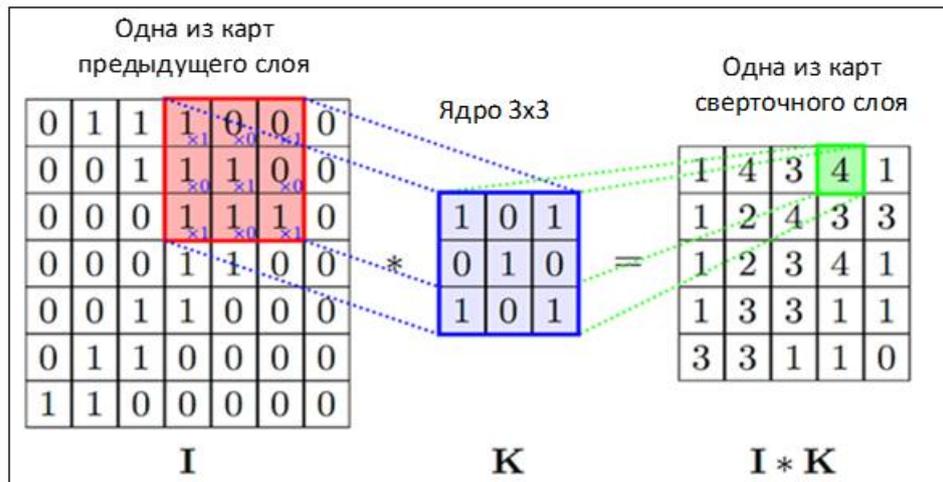


Рисунок 15 – Операция свертки и получение значений карты признаков

1.3.3 Подвыборочный слой

Подвыборочный слой также, как и сверточный имеет карты, но их количество совпадает с предыдущим (сверточным) слоем. Цель использования данного слоя – уменьшить размерности карт предыдущего слоя. Если на предыдущей операции свертки уже были выявлены некоторые признаки, то для дальнейшей обработки настолько подробное изображение уже не нужно, и оно уплотняется до менее подробного. К тому же фильтрация уже ненужных деталей помогает не переобучаться.

В процессе сканирования ядром подвыборочного слоя (фильтром) карты предыдущего слоя, сканирующее ядро не пересекается в отличие от сверточного слоя. Обычно, каждая карта имеет ядро размером 2x2, что позволяет уменьшить предыдущие карты сверточного слоя в 2 раза. Вся карта признаков разделяется на ячейки 2x2 элемента, из которых выбираются максимальные по значению. Обычно в подвыборочном слое применяется функция активации (ReLU, Rectifiedlinearunit). Операция подвыборки выполняется (Max-Pool – выбор максимального) в соответствии с рисунком 6.

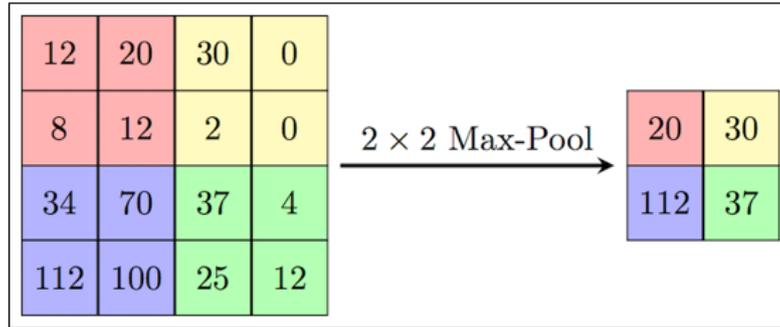


Рисунок 16. Формирование новой карты подвыборочного слоя на основе предыдущей карты сверточного слоя. Операция подвыборки (MaxPooling)

Слой может быть описан формулой:

$$x^l = f(a^l \cdot \text{subsample}(x^{l-1}) + b^l), \quad (4)$$

где x^l – выход слоя l , f – функция активации, a^l , b^l – коэффициенты сдвига слоя l , subsample – операция выборки локальных максимальных значений.

1.3.4 Полносвязный слой

Последний из типов слоев – это слой обычного многослойного персептрона. Цель применения данного слоя – это обращение к выходу предыдущего слоя и определение свойств, которые больше связаны с определенным классом.

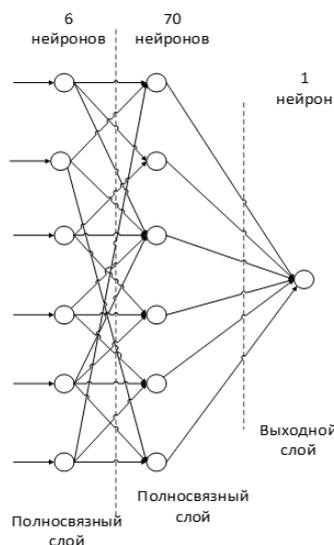


Рисунок 17 – Пример полносвязных слоев

Нейроны каждой карты предыдущего подвыборочного слоя связаны с одним нейроном скрытого слоя. Таким образом число нейронов скрытого слоя равно числу карт подвыборочного слоя, но связи могут быть не обязательно такими, например, только часть нейронов какой-либо из карт подвыборочного слоя связана с первым нейроном скрытого слоя, а оставшаяся часть со вторым, либо все нейроны первой карты связаны с нейронами 1 и 2 скрытого слоя.

1.4 Обзор программных инструментов для создания веб-сервиса

Веб-сервис – это стандартизированная среда для установления связи между клиентскими и серверными приложениями во всемирной паутине (World Wide Web). Веб-сервис является программным модулем, предназначенным для выполнения определенного набора задач, такие как взаимодействие между сайтами, обмен данными между различными приложениями, которые написаны даже на различных языках и распределены на разных узлах сети, взаимодействие с пользователями [3].

Для того чтобы веб-служба была полностью функциональна, необходимо наличие определенных компонентов. Эти компоненты должны присутствовать независимо от того, какой язык разработки используется для программирования [4]. Основным компонентом веб-сервиса являются данные, которые передаются между клиентом и сервером. Данные могут размечены с использованием *языка разметки* (XML), который является аналогом HTML, понятен для промежуточного языка.

В данной работе в качестве программных инструментов для разработки веб-сервиса мы использовали следующие языки программирования.

Язык гипертекстовой разметки (HyperText Markup Language, HTML) – простой кроссплатформенный язык разметки, используемый для создания веб-сайтов, гипертекстовых документов и документов на веб-странице.

HTML создает систему для идентификации и описания различных компонентов документа (заголовки, абзацы, таблицы, списки), а также может представлять гипертекстовые новости, почту, документацию и гипермедиа (меню опций, результаты запроса к базе данных), простые структурированные документы со встроенной графикой, гипертекстовые представления существующих массивов информации [5].

Каскадные таблицы стилей (CSS) – определяют внешний вид веб-страницы, представленной для пользователей. С помощью CSS можно легко и быстро управлять шрифтами, интервалами между строками, цветами, фоновыми изображениями, макетами и прочими атрибутами оформления элементов веб-сайта [6]. Каскадные таблицы стилей – это инструмент, позволяющий автоматизировать процесс разработки, а также управлять представлением документов не только в браузерах, но при их печати. Кроме того, в таблицах стилей существуют правила для определения не визуального представления документов, например, как будет звучать текст документа при его программном считывании.

JavaScript – интерпретируемый, мультипарадигменный объектно-ориентированный язык программирования, который обеспечивает динамический, объектно-ориентированный, императивный и функциональный стили программирования. Он является кроссплатформенным и поддерживается большинством веб-браузеров, например, Chrome, Firefox, Internet Explorer, Opera, Safari, Edge и большинство мобильных браузеров и смартфонов [6, 7]. Язык JavaScript в основном используется как язык сценариев для управления элементами веб-страницы, придавая им интерактивность и вариативность поведения:

- создание интерфейсных виджетов, раскрывающихся меню,
- проверка корректности значений, введенных в элементы формы,
- создание обработчиков событий, которые отвечают за пользовательские и другие события,

- обеспечивает динамику и изменение стилей оформления элементов графических элементов интерфейса,
- выполнение требований от браузера (например, запомнить информацию о пользователе).

Django – веб-фреймворк, представляет собой набор модульных инструментов, которые помогают разработчику абстрагироваться от многих формальностей и рутинных операций, присущих веб-разработке. Например, большинству веб-сайтов требуется одинаковая базовая функциональность: возможность подключаться к базе данных, задавать маршруты URL-адресов, отображать содержимое на странице, правильно управлять безопасностью и т.д.

Фреймворк Django реализован на языке Python и включает в себя поддержку решения общих задач веб-разработчика:

- аутентификация пользователя,
- шаблоны, маршруты и представления,
- интерфейс администратора,
- обеспечение надежной безопасности,
- поддержка нескольких баз данных.

В качестве основного программного инструмента, был выбран язык программирования Python. Этот язык программирования является кроссплатформенным и позволяет работать быстрее и более эффективно интегрировать в разных операционных системах. Более того, Python широко применяется в сфере машинного обучения, анализ больших данных, компьютерного зрения и т.д.

В настоящее время существует несколько современных фреймворков для работы с изображениями, поддерживающими глубокое обучение:

– **TensorFlow** [24] – открытая программная библиотека для машинного обучения, разработанная компанией Google для решения задач построения и

тренировки нейронной сети с целью автоматического нахождения и классификации образов, достигая качества человеческого восприятия [24].

– **OpenCV** [22] – библиотека с открытым исходным кодом, используется для всех видов обработки и анализа изображений и видео, которая включает в себя несколько сотен алгоритмов компьютерного зрения: интерпретации изображений, калибровки камеры по эталону, устранение оптических искажений, определение сходства, анализ перемещения объекта, определение формы объекта и слежение за объектом, 3D-реконструкция, сегментация объекта, распознавание жестов и т.д.

– **Keras** [1] – это минималистичная открытая нейросетевая библиотека, написанная на языке Python для глубокого обучения, которая может работать поверх TensorFlow. Данная библиотека нацелена для оперативной работы с сетями глубокого обучения, при этом спроектирована так, чтобы быть компактной, модульной и расширяемой.

– **NumPy**[1] – это библиотека, с открытым исходным кодом для языка программирования Python, с поддержкой многомерных массивов (включая матриц) и поддержкой высокоуровневых математических функций, предназначенных для работы с многомерными массивами.

2.1 Сравнение характеристик систем распознавания объектов

В таблице 1 приведены сравнительные характеристики существующих программных фреймворков для распознавания объектов.

Из таблицы 1 видно, что указанные фреймворки имеют ряд плюсов и минусов. Большинство из них разрабатывались для определенных задач, и они практически недоступны для конечных пользователей из-за высокой коммерческой стоимости. На этом фоне выигршно отличается YOLO(YouOnlyLookOnce), у которой программный код открыт, но быстроедействие отставляет желать лучшего. Фреймворк Safe (использован в

данной работе) после обучения способен распознавать объектов более точно, за приемлемое время.

Таблица 1 – Сравнительные характеристики

Фреймворк	Средняя точность обнаружения объектов (%)	Количество кадров в секунду	Обучаемость	Время обнаружения (сек)	Исходный код
Fast R-CNN	86.4	7	-	7	Закрыт
RetinalNet	82	12	-	5	Закрыт
R-FCN	77.6	16	-	4	Закрыт
YOLO	81.4	20	+	6	Открыт
Cafe	84.6	28	+	3	Открыт

1.5 Постановка задачи

На основании проведенного литературного обзора сформулирована цель работы: применить алгоритмы и методы искусственных нейронных сетей для анализа изображения и видеоинформации для распознавания 3D-объектов.

Для достижения поставленной цели реализованы следующие задачи:

- 1) Выбор языков программирования, подготовка рабочей среды для реализации выбранных алгоритмов и методов.

- 2) Практическое применение алгоритмов методов классификация изображений для распознавания объектов.
- 3) Выбор оптимальных параметров используемых моделей.
- 4) Проведение машинного обучения на тренировочных выборках.
- 5) Проведение тестирования и анализ полученных результатов.

2. Результаты проведенного исследования

В данной работе были использованы основные алгоритмы и методы для распознавания 3D-объектов и применили на практике основные операции над изображениями: фильтрация, обнаружение контуров, преобразование уровней яркости. В качестве основного языка программирования был выбран язык программирования Python, были настроены и импортированы необходимые библиотеки: NumPy [2], Matplotlib[2], OpenCV [22], TensorFlow [24], Caffe [23].

Для обучения нашей модели были использованы набор данных CIFAR-10[26], который содержит 60 000 цветных изображений размера 32×32 пикселя с 3 каналами, разбитых на 10 классов.

В нашей нейронной сети была применено 32 сверточных фильтра размера 3×3 . Размер выхода такой же, как размер входа, т. е. 32×32 , а в качестве функции активации использовали функцию ReLU. Далее следует операция max-пулинга с размером блока 2×2 и прореживание с коэффициентом 25%. Ниже представлена модель, которую мы использовали для реализации данного проекта.

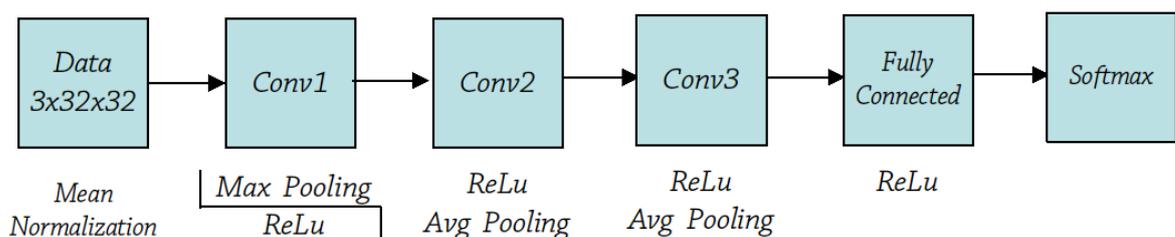


Рисунок -18. Модель сверточной нейронной сети для распознавания

В предложенной модели мы использовали первый слой, который преобразует входное изображение на 3 канала для проведения последующих операций наследующих слоях. Следующий этап это max-пулинг и применение функции активации ReLU. Далее производится фильтрация и обнаружение ключевых признаков объекта. Конечный слой имеет функцию

активации softmax, с помощью которого определяются класс объекта. Код представлен в Приложении 2.

2.2 Алгоритмы для распознавания объектов

В дальнейшей работе мы в качестве входных данных использованы изображения в формате *.jpg и *.png.

Преобразование уровня яркости. Встречаются изображения, на которых интересующий нас объект где из-за плохого освещения находится в более темном месте, что затрудняет его выявление. Для решения указанной проблемы мы использовали метод преобразование уровня яркости. Прочитав входного изображение в виде матрицы с помощью NumPy, мы можем применить к нему различные математические операции, в данном случае выравниваем гистограмму.

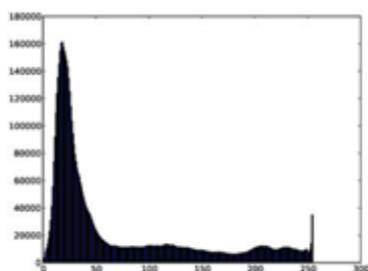
Функция принимает полутоновое изображение и количество интервалов в гистограмме, и возвращает изображение, для которого гистограмма выровнена с помощью функция распределения (рисунок 19).



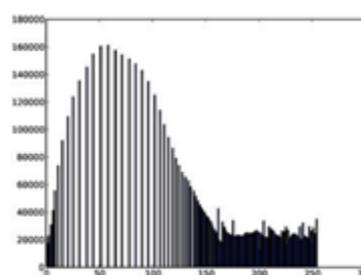
а)



б)



в)



г)

Исходя из результатов видно, что контрастность увеличилась, и детали в темных участках теперь видны более отчетливо.

Фильтрация контуров объектов. Когда объект достаточно сложный, но изображение достаточно контрастное, то зачастую единственным способом работы с ним является выделение его контуров[18]. В данной работе мы использовали алгоритм обнаружения границ Кэнни. Результаты показаны на рисунке 20. Код этого примера представлен в Приложении 1.



Рисунок 20 – а) Группа однотипных объектов на переднем плане

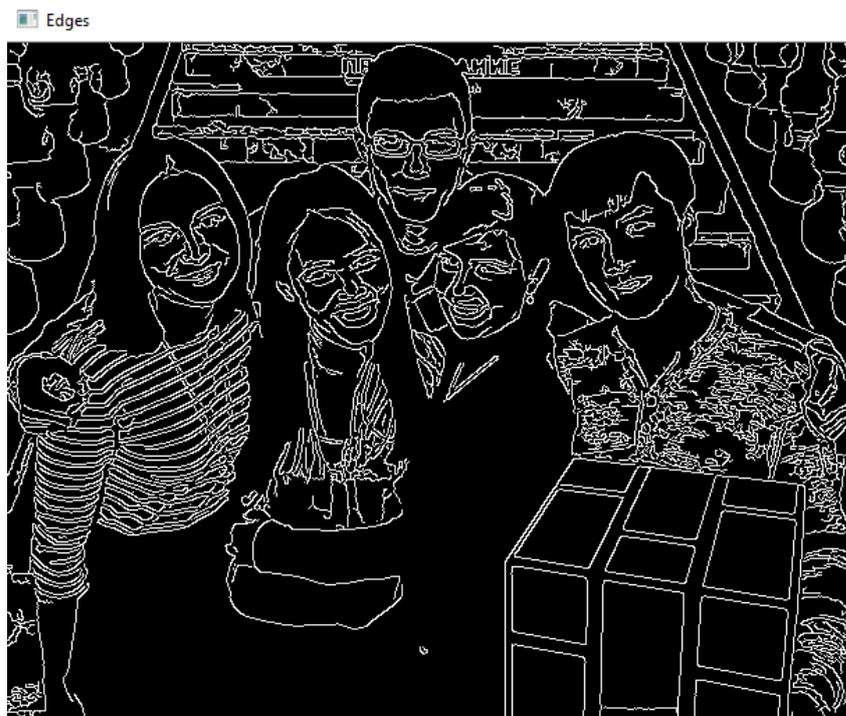


Рисунок 20 – б) Обнаруженные границы объектов

Исходя из результата видно, что основные границы объектов на изображение найдены с удовлетворительной точностью, и его можно использовать в дальнейшей работе в качестве дополнительной информации о структуре объектов. Если нейросеть имеет дополнительную информацию об объекте, с которыми она работает, то качество распознавания будет только возрастать.

2.3 Распознавание 3D – объектов

Для анализа входных данных[18] с использованием искусственных нейронных сетей, мы будем использовать библиотеку OpenCV и NumPy и модель Caffe, реализованная на Keras и TensorFlow. В качестве среды разработки и реализации поставленной задачи, была выбрана интегрированная среда PyCharm.

Для распознавания, рассмотрим примеры простых объектов в естественном виде. Результат показан на рисунке 21. Код данного примера представлен в Приложении 2.



Рисунок 21 – а) объект (самолёт), б) обнаруженный объект с указанием точностью распознавания

Рисунок 21 показывает, что обученная нейронная сеть может распознавать простые объекты с высокой точностью, в данном случае точность распознавания составляет 99.36%.

Теперь проверим точность распознавания для спроектированной нейронной сети на более сложных объектах (рисунок 22).

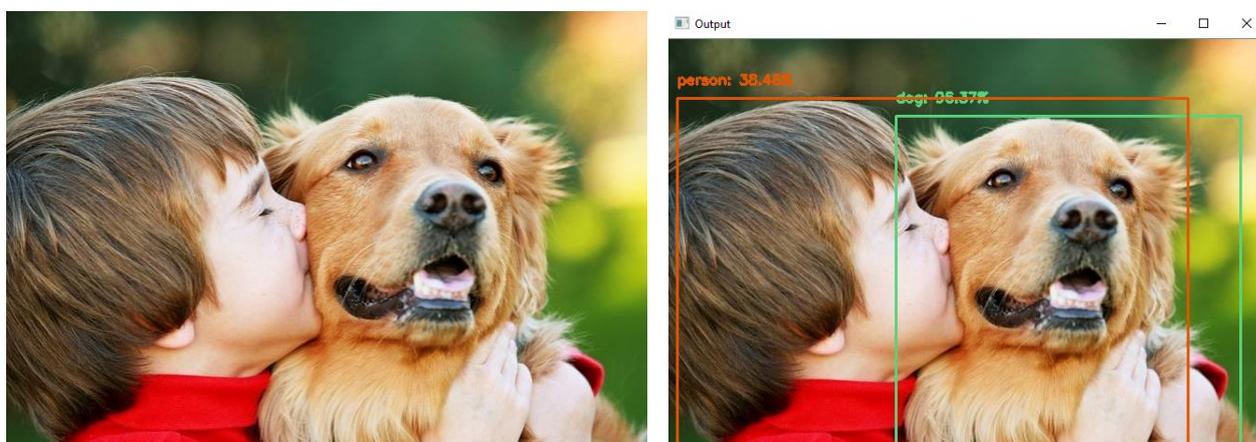


Рисунок 22а – а) исходное изображение, б) обнаруженные объекты с указанием точностью распознавания

Полученный нами результат по распознаванию объектов на изображениях показывает точность распознавания: собака(dog) –96.37%, человек (person) – 38.46%.

```
C:\Users\Anzor\AppData\Local\Programs\  
[INFO] loading model...  
[INFO] computing object detections...  
[INFO] dog: 96.37%  
[INFO] person: 38.46%
```

Рисунок 23 – Результат выявления объектов с указанием точности

Нейронная сеть распознала два объекта объектов, но объект 1 (ребенок) распознан с существенно более низким качеством, на точность решения оказало влияние неестественное положение объекта (поворот на 90 градусов).

Рассмотрим более распространённые ситуации с более естественным положением объектов. На изображении присутствует несколько однотипных объектов (рисунок 24).



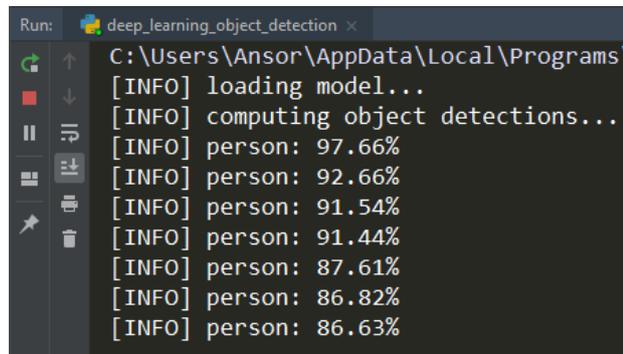
Рисунок 24 – а) восемь однотипных объектов



Рисунок 24 – б) результат обнаружения семи объектов с указанием точности

Полученный результат показывает, что объекты обнаружены с высоким процентом точности (86% и более), однако один объект не был идентифицирован.

Ниже представлены результаты вычисления программы, количество обнаруженных объектов с указанием точности решения.



```
Run: deep_learning_object_detection x
C:\Users\Anzor\AppData\Local\Programs\
[INFO] loading model...
[INFO] computing object detections...
[INFO] person: 97.66%
[INFO] person: 92.66%
[INFO] person: 91.54%
[INFO] person: 91.44%
[INFO] person: 87.61%
[INFO] person: 86.82%
[INFO] person: 86.63%
```

Рисунок 25 – Результат вычисления и выявления объектов.

Исходя из выше представленных результатов тестирования обученной нейронной сети, делаем вывод, что обнаружения и распознавания объектов происходит с достаточно высокой точностью.

3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» бакалаврской работе является: образовательная онлайн –платформа для школьников, отвечающего современным требованиям в области ресурсоэффективности и ресурсосбережения.

Для реализация поставленной цели, в данном разделе производится расчет экономической оценки создания, реализации и тестирования алгоритмов распознавания объектов, для чего необходимо:

- реализовать серверную часть разрабатываемого продукта.
- реализовать клиентскую часть разрабатываемого продукта.
- Разработать календарный план работ.
- Рассчитать затраты на материалы.
- Рассчитать заработную плату исполнителей.
- Рассчитать затраты на электроэнергию.
- Рассчитать амортизационные расходы.
- Рассчитать накладные расходы.
- Определить затрат на реализацию проекта.
- Оценить экономическую эффективность.

3.1. Организация и планирование работ

Организация и планирования проекта необходима для того, чтобы правильно спланировать занятость участников проекта и сроки проведения работ. В таблице 3.1 представлено планирование загрузки каждого исполнителя проекта для каждого этапа работ. Загрузка исполнителей представлена в процентах, что представляет собой рациональную продолжительность. Этапы работ упорядочены в хронологическом порядке.

Таблица 3.1

Перечень работ и продолжительность их выполнения.

Этапы работы	Исполнители	Загрузка исполнит
Постановка целей и задач, получение исходных данных	НР	НР – 100%
Составление и утверждение ТЗ	НР, И	НР – 100% И – 50%
Подбор и изучение материалов тематике	НР, И	НР – 30% И – 100%
Разработка календарного плана	НР, И	НР – 100% И – 50%
Обсуждениелитературы	НР, И	НР – 20% И – 100%
Выбор СУБД и проектирование БД	НР, И	НР – 100% И – 80%
Разработка веб – сайта	НР, И	НР – 70% И – 100%
Получение и обработка данных	И	И – 100%
Оформление расчетно-пояснительной записи	И	И – 100%
Оформление графического материала	И	И – 100%
Подведение итогов	НР, И	НР – 60% И – 100%

Примечание к Таблице 3.1: И – исполнитель; НР – научный руководитель

3.1.1. Продолжительность этапов работ

Существует несколько методов для расчета продолжительности этапов работ, например, технико-экономический и опытно-статистический. Первый из них применяется для случаев с хорошо развитой нормативной базой трудоемкости задач. Это обычно связано с их частой повторяемостью. Однако, это почти невозможно в большинстве случаев, и поэтому чаще используется опытно-статистический метод, реализуемый как аналоговый или экспертный.

В аналоговом случае имеем достаточно простой и не затратный способ, возможный лишь тогда, когда в поле досягаемости имеется 39 неустаревающий аналог того, что планируется выполнить в ходе выпускной квалификационной работы. Но в большинстве случаев его глобальное применение невозможно, однако, он отлично применяется локально, для небольших частей работы. Экспертный способ предусматривает привлечение экспертов предметной области для оценки трудоемкости каждого этапа работ. В таблице 3.2 представлена продолжительность каждого этапа работ и занятость каждого исполнителя. Столбцы с 3 по 5 содержат реализацию экспертного подхода.

Таблица 3.2

Трудозатраты на выполнение проекта

Этап	Исполнители	Продолжительность работ, дни			Трудоемкость работ по исполнителям чел.- дн.			
					Т _{рд}		Т _{кд}	
		t_{min}	t_{max}	$t_{ож}$	НР	И	НР	И
Постановка задачи	НР	2	4	2,8	3,36	0	4,05	0
Разработка и утверждение технического задания (ТЗ)	НР, И	3	4	3,4	3,67	0,41	4,42	4,49
Подбор и изучение материалов по тематике	НР, И	9	13	8,6	4,75	15,84	5,75	19,16
Разработка календарного плана	НР, И	3	6	4,2	4,03	1,01	4,86	1,22
Обсуждение литературы	НР, И	3	6	4,2	2,52	2,52	3,04	3,04
Выбор СУБД и проектирование БД	НР, И	12	18	12,6	7,6	15,1	9,1	18,2
Разработка веб – сайта	НР, И	10	14	11,6	9,7	13,9	11,7	16,8
Получение и обработка данных	И	12	16	13,6	0	16,3	0	19,7
Оформление расчетно-пояснительной записки	И	11	15	11,6	0	13,9	0	16,8
Оформление графического материала	И	7	9	7,8	0	9,4	0	11,3
Подведение итогов	НР, И	5	7	5,8	4,2	7	5	8,4
Итого:		77	112	86,2	39,83	95,38	47,92	119,11

Таблица 3.3

Линейный график работ (пример)

Этап	Н	С	Март			Апрель			Май			Июнь	
			10	20	30	40	50	60	70	80	90	100	110
1	4,05	–	■										
2	4,42	4,49	■										
3	5,75	19,16		■									
4	4,86	1,22		■									
5	3,4	3,2				■							
6	9,1	18,2				■							
7	11,7	16,8				■							
8	–	19,7						■					
9	–	16,8						■					
10	–	11,3						■					
11	5	8,4											■

НР – ■; И – ■

3.2. Расчет смены затрат на выполнение проекта

В состав затрат на создание проекта включается величина всех расходов, необходимых для реализации комплекса работ, составляющих содержание данной разработки. Расчет сметной стоимости ее выполнения производится по следующим статьям затрат:

- материалы и покупные изделия;
- заработная плата;
- социальный налог;
- расходы на электроэнергию (без освещения);
- амортизационные отчисления;
- командировочные расходы;
- оплата услуг связи;
- арендная плата за пользование имуществом;
- прочие услуги (сторонних организаций);
- прочие (накладные расходы) расходы.

3.2.1. Расчет затрат на материалы

Затраты на материалы включают в себя стоимость всех материалов, необходимых для выполнения работ, например, стоимость материалов, каких-либо покупных изделий, оборудование, стоимость которого не превышает 40 тысяч рублей. Для определения стоимости каких-либо материалов или оборудования, необходимо посмотреть ее в договоре поставки или в другом сопроводительном документе.

Также в затраты на материалы обычно включаются расходы на различные операции купли-продажи. По-другому эти операции называются транзакциями. Обычно они оцениваются приблизительно как 5-15 процентов

от цены 44 оборудования или материала. Расчет затрат на материалы представлен в таблице 4.4.

Таблица 3.4

Расчет затрат на материалы

Наименование материалов	Цена за ед., руб.	Кол-во	Сумма, руб.
Бумага для принтера формата А4	250	1 уп.	250
Картридж для принтера	2500	1 шт.	2500
Канцелярские принадлежности	300	1 шт.	300
Итого:			3050

Допустим, что ТЗР составляют 8 % от отпускной цены материалов, тогда расходы на материалы с учетом ТЗР равны $C_{\text{мат}} = 3050 * 1,08 = 3314$ руб.

3.2.2. Расчет заработной платы

Здесь приводится описание расходов на зарплаты научного руководителя и исполнителя. Все расчеты выполняются, учитывая трудоемкость выполнения каждого этапа и оклад исполнителя в месяц. Для расчета среднедневной тарифной заработной платы, можно воспользоваться следующей формулой:

$$ЗП_{\text{дн-т}} = MO/24.83 \quad (3.1)$$

Она учитывает то, что всего в году 298 рабочих дней, а в месяце соответственно

$$298 / 12 = 24.83. \text{ Здесь рассматривается шестидневная рабочая неделя.}$$

В таблице 3.5 приведены расчеты заработной платы исполнителя (инженера) и научного руководителя. Здесь, из таблицы 3.2 взяты временные затраты по каждому исполнителю проекта (данные взяты в рабочих днях путем округления до ближайшего целого). Для того, чтобы учесть в составе заработной платы премии и дополнительные заработные платы, а также

районные надбавки, будем использовать следующие коэффициенты: КПП = 1,1; Кдоп.ЗП = 1,188; Кр = 1,3. Таким образом, для перехода от базовой (тарифной) суммы заработной платы исполнителя, связанной с участием в проекте, к соответствующей зарплатной части сметы (полному заработку) необходимо первую умножить на интегральный коэффициент $K_i = 1,3 * 1,188 * 1,1 = 1,7$. Значение Кдоп. ЗП, упомянутое выше, применяется при шестидневной рабочей неделе. В случае пятидневной рабочей недели оно равно 1,113 и в этом случае интегральный коэффициент $K_i = 1,62$.

Таблица 3.5

Затраты на заработную плату

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб./раб. день	Затраты времени, раб. дни	Коэффициент	Фонд з/платы, руб.
НР	33 664	1342,09	39	1,699	88457
С	9489	533	95	1,62	81016
Итого:					169473

3.2.3. Расчет затрат на социальный налог

Затраты на единый социальный налог (ЕСН), включающий в себя отчисления в пенсионный фонд, на социальное и медицинское страхование, составляют 30 % от полной заработной платы по проекту, т.е. $C_{соц.} = C_{зп} * 0,3$. Итак, в нашем случае $C_{соц.} = 169473 * 0,3 = 50842$ руб.

3.2.4. Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию, потраченную в ходе выполнения проекта на работу используемого оборудования, рассчитываемые по формуле:

$$C_{эл.об.} = P_{об} \cdot t_{об} \cdot Ц_{э} \quad (3.2)$$

где $P_{\text{ОБ}}$ – мощность, потребляемая оборудованием, кВт;

ЦЭ – тариф на 1 кВт·час;

$t_{\text{об}}$ – время работы оборудования, час.

Для ТПУ $\text{ЦЭ} = 5,748$ руб./кВт·час (с НДС).

Время работы оборудования вычисляется на основе итоговых данных таблицы 3.2 для инженера (ТРД) из расчета, что продолжительность рабочего дня равна 8 часов.

$$t_{\text{об}} = T_{\text{РД}} * K_t, \quad (3.3)$$

где $K_t \leq 1$ – коэффициент использования оборудования по времени, равный отношению времени его работы в процессе выполнения проекта к ТРД, определяется исполнителем самостоятельно. В ряде случаев возможно определение $t_{\text{об}}$ путем прямого учета, особенно при ограниченном использовании соответствующего оборудования.

Мощность, потребляемая оборудованием, определяется по формуле:

$$P_{\text{ОБ}} = P_{\text{ном.}} * K_C \quad (3.4)$$

где $P_{\text{ном.}}$ – номинальная мощность оборудования, кВт;

$K_C \leq 1$ – коэффициент загрузки, зависящий от средней степени использования номинальной мощности. Для технологического оборудования малой мощности $K_C = 1$.

Пример расчета затраты на электроэнергию для технологических целей приведен в таблице 3.6.

Таблица 3.6

Затраты на электроэнергию технологическую

Наименование оборудования	Время работы оборудования $t_{\text{об}}$, час	Потребляемая мощность $P_{\text{об}}$, кВт	Затраты $\text{Э}_{\text{об}}$, руб.
Персональный компьютер	$98.38 * 8 = 787,04$	0,33	1492,9

Струйный принтер	10	0,4	20,8
Итого:			1513,7

3.2.5. Расчет амортизационных расходов

В статье «Амортизационные отчисления» рассчитывается амортизация используемого оборудования за время выполнения проекта.

Используется формула

$$C_{AM} = \frac{N_A * C_{OB} * t_{pf} * n}{F_D}, \quad (4.5)$$

где N_A – годовая норма амортизации единицы оборудования;

C_{OB} – балансовая стоимость единицы оборудования с учетом ТЗР. При невозможности получить соответствующие данные из бухгалтерии она может быть заменена действующей ценой, содержащейся в ценниках, прейскурантах и т.п.;

F_D – действительный годовой фонд времени работы соответствующего оборудования, берется из специальных справочников или фактического режима его использования в текущем календарном году. При этом второй вариант позволяет получить более объективную оценку C_{AM} . Например, для ПК в 2019 г. (298 рабочих дней при шестидневной рабочей неделе) можно принять $F_D = 298 * 8 = 2384$ часа;

$t_{рф}$ – фактическое время работы оборудования в ходе выполнения проекта, учитывается исполнителем проекта;

n – число задействованных однотипных единиц оборудования.

При использовании нескольких типов оборудования расчет по формуле делается соответствующее число раз, затем результаты суммируются.

Для определения N_A следует обратиться к приложению 1, содержащему фрагменты из постановления правительства РФ «О классификации основных средств, включенных в амортизационные группы». Оно позволяет получить рамочные значения сроков амортизации (полезного использования) оборудования $\equiv C_A$. Например, для ПК это $2 \div 3$ года. Необходимо задать конкретное значение C_A из указанного интервала, например, 2,5 года. Далее определяется N_A как величина обратная C_A , в данном случае это $1 : 2,5 = 0,4$.

Стоимость ПК 45000 руб, время использования 98,38 часа, тогда для него $C_{AM}(ПК) = (0,4 * 45000 * 98,38 * 1) / 2408 = 735,4$ руб. Стоимость принтера 12000 руб., его $F_D = 500$ час.; $N_A = 0,5$; $t_{рф} = 30$ час., тогда его $C_{AM}(Пр) = (0,5 * 12000 * 30 * 1) / 500 = 360$ руб.

Итого начислено амортизации 6 364,02 руб.

3.2.6. Расчет прочих расходов

Для вычисления прочих расходов, необходимо их принять как 10% от суммы всех остальных расходов.

$$\text{Спроч.} = (3294 + 169473 + 50842 + 1513,7 + 6364,02) \cdot 0,1 = 23148,7 \text{ руб.}$$

3.2.7. Расчет общей себестоимости разработки

Общая себестоимость проекта по категориям представлена в таблице 3.7.

Таблица 3.7

Смета затрат на разработку проекта

Статья затрат	Условное обозначение	Сумма, руб.
Материалы и покупные изделия	$C_{\text{мат}}$	3294
Основная заработная плата	$C_{\text{зп}}$	169473
Отчисления в социальные фонды	$C_{\text{соц}}$	50842
Расходы на электроэнергию	$C_{\text{эл.}}$	1513,7
Амортизационные отчисления	$C_{\text{ам}}$	6364,02
Прочие расходы	$C_{\text{проч}}$	23148,7
Итого:		254635,42

Таким образом, затраты на разработку составили $C = 254635,42$ руб.

3.2.8. Расчет прибыли

Прибыль от реализации проекта в зависимости от конкретной ситуации (масштаб и характер получаемого результата, степень его определенности и коммерциализации, специфика целевого сегмента рынка и т.д.) может определяться различными способами. Если исполнитель работы не располагает данными для применения «сложных» методов, то прибыль следует принять в размере $5 \div 20$ % от полной себестоимости проекта. В

нашем примере она составляет 50927,1 руб. (20 %) от расходов на разработку проекта.

3.2.9. Расчет НДС

НДС составляет 20% от суммы затрат на разработку и прибыли. В нашем случае это $(254635,42 + 50927,1) * 0,2 = 61112,5$ руб.

3.2.10. Цена разработки НИР

Цена равна сумме полной себестоимости, прибыли и НДС, в нашем случае

$$C_{\text{НИР(КР)}} = 254635,42 + 50927,1 + 61112,5 = 366675,021 \text{руб.}$$

3.3. Оценка экономической эффективности проекта

Оценка экономического эффекта выходит за рамки представленной работы и предполагает отдельного системного исследования. Результаты данной работы по разработке веб-сервиса по распознаванию 3D-объектов на основе сверточных нейронных сетей могут быть использованы в коммерческих и личных целях. Применять такой проект можно для автоматизации распознавания транспорта и аппаратов в производственном помещении, а также в интернет-магазинах для быстрого нахождения интересующего продукта внутри этой сети.

Это позволяет отказаться от водителя или постороннего наблюдателя для управления транспортов и аппаратов в производственном помещении, что напрямую уменьшает расходы на персонал, а если принять этого проекта и взаимодействовать с интернет-магазинов, покупатели могут в несколько раз сэкономить время и удобно выполнять покупку. Также может быть эффективным применение этой технологии в других сферах и условиях, где необходимо распознавать 3D-объекты. Количественная оценка экономической эффективности невозможна в рамках данной работы, так как не задана конкретная область его применения.

4. Социальная ответственность

В настоящее время большое внимание уделяется вопросам безопасности трудящихся на рабочем месте. Одной из основных задач является охрана здоровья сотрудников предприятий, сведение к минимуму или ликвидация различных видов производственных травм и снижение риска заболеваний. Такие факторы рассматривает социальная ответственность, а именно: состояние рабочего места, помещения, режим работы, обеспечение мероприятий по защите трудящихся в случае возникновения чрезвычайной ситуации. Существует международный стандарт ИСО 26000 «Руководство по социальной ответственности», разработанный в 2012 г., в соответствии с которым осуществляется разработка системы социальной ответственности на предприятии. При применении организацией данного стандарта для разработки системы социальной ответственности, она решает ряд проблем, таких как: проблемы, связанные с потребителями, трудовые практики, участие в жизни сообществ и их развитие, права человека, окружающая среда, добросовестные деловые практики.

В современной жизни компьютер широко применяется в повседневную жизнь людей и их использование постоянно увеличивается. Несоблюдение требований безопасности персональных компьютеров приводит к тому, что при работе за компьютером пользователь может ощущать дискомфорт: возникают головные боли и резь в глазах, появляются усталость и раздражительность.

Цель выпускной квалификационной работы заключается в разработке веб-сервиса по распознаванию 3D – объектов на основе сверточных нейронных сетей, также рассмотрены некоторые методы исследования на изображениях с помощью существующих алгоритмов.

В данной части бакалаврской работе рассмотрен анализ вредных и опасных факторов производства, методы их предупреждения, организация

мероприятий защиты в случае возникновения чрезвычайной ситуации. Для эффективного ведения разработки обозначенной системы необходимо учитывать социальные, правовые и экологические вопросы охраны здоровья и безопасности жизнедеятельности.

В рамках данного раздела необходимо:

- определить и изучить возможные вредные факторы, влияющие на исполнителей при разработке программного продукта;
- разработать решения для минимизации их влияния;
- произвести анализ разработанных решений, с точки зрения социальной ответственности за моральные, общественные, экономические и экологические возможные негативные последствия и ущерб здоровью человека в результате их внедрения.

4.1 Правовые и организационные вопросы обеспечения безопасности

Для обеспечения высокопроизводительного труда необходимо создать для работника благоприятные условия труда.

Условия труда – это совокупность факторов производственной среды и трудового процесса, оказывающих влияние на здоровье и работоспособность человека в процессе труда.

Регулирование отношений между работником и работодателем, касающихся оплаты труда, трудового распорядка, особенности регулирования труда женщин, детей, людей с ограниченными способностями и проч., осуществляется законодательством РФ, а именно трудовым кодексом РФ [27].

Нормальная продолжительность рабочего времени не может превышать 40 часов в неделю. Порядок исчисления нормы рабочего времени на определенные календарные периоды (месяц, квартал, год) в зависимости от установленной продолжительности рабочего времени в неделю определяется

федеральным органом исполнительной власти, осуществляющим функции по выработке государственной политики и нормативно-правовому регулированию в сфере труда.

В течение рабочего дня (смены) работнику должен быть предоставлен перерыв для отдыха и питания. Время предоставления перерыва и его конкретная продолжительность устанавливаются правилами внутреннего трудового распорядка или по соглашению между работником и работодателем.

Всем работникам предоставляются выходные дни (еженедельный непрерывный отдых).

Организация-работодатель выплачивает заработную плату работникам. Возможно удержание заработной платы только в случаях, установленных ТК РФ ст. 137. В случае задержки заработной платы более чем на 15 дней, работник имеет право приостановить работу, письменно уведомив работодателя.

Законодательством РФ запрещена дискриминация по любым признакам и принудительный труд.

Если пользователь постоянно загружен работой с ЭВМ, приемлемой является поза сидя. В положении сидя основная нагрузка падает на мышцы, поддерживающие позвоночный столб и голову. В связи с этим при длительном сидении время от времени необходимо сменять фиксированные рабочие позы.

Исходя из общих принципов организации рабочего места, в нормативно-методических документах сформулированы требования к конструкции рабочего места. Основными элементами рабочего места программиста являются: рабочий стол, рабочий стул (кресло), монитор, клавиатура, мышь; вспомогательными - пюпитр, подставка для ног. Взаимное расположение элементов рабочего места должно обеспечивать возможность осуществления

всех необходимых движений и перемещений для эксплуатации и технического обслуживания оборудования. Рабочее место при выполнении работ в положении сидя должно соответствовать требованиям ГОСТ 12.2.032-78.

Рабочие места с ЭВМ должны располагаться на расстоянии не менее 1,5 метра от стены с оконными проемами, от других стен – на расстоянии 1 м, между собой – на расстоянии не менее 1,5 м. При размещении рабочих мест необходимо исключить возможность прямой засветки экрана источником естественного освещения.

Располагать ЭВМ на рабочем месте необходимо так, чтобы поверхность экрана находилась на расстоянии 400 – 700 мм от глаз пользователя. Конструкция рабочего места и взаимное расположение всех его элементов (сиденье, органы управления, средства отображения информации и т.д.) должны соответствовать антропометрическим, физиологическим и психологическим требованиям, а также характеру работы. Рабочее кресло обеспечивает поддержание рабочей позы в положении сидя, и чем длительнее это положение в течение рабочего дня, тем жестче должны быть требования к созданию удобных и правильных рабочих сидений. Высота поверхности сиденья должна регулироваться в пределах 400 – 550 мм. При организации рабочего пространства необходимо учитывать индивидуальные антропометрические параметры пользователя с соответствующими допусками на возможные изменения рабочих поз и потребность в перемещениях.

Рациональной рабочей позой может считаться такое расположение тела, при котором ступни работника расположены на плоскости пола или на подставке для ног, бедра сориентированы в горизонтальной плоскости, верхние части рук – вертикальный угол локтевого сустава колеблется в пределах 70 – 90, запястья согнуты под углом не более чем 20, наклон головы – в пределах 15 – 20, а также исключены частые ее повороты.

Продолжительность непрерывной работы с компьютером без регламентированного перерыва не должна превышать 2-х часов. Это связано с тем, что на человека при работе с компьютером оказывают влияние опасные и вредные производственные факторы, а также наступает общее утомление, что негативно сказывается на здоровье и самочувствии человека.

Электроснабжение сети переменного напряжения 220В. Помещение без повышенной опасности в отношении поражения человека электрическим током по ГОСТ 12.1.013-78. Компьютер, расположенный на рабочей поверхности высотой 770 мм., обладает следующими характеристиками: процессор Intel Core i5-3230m, оперативная память 8 ГБ, система Microsoft Windows 10, частота процессора – 2,6 ГГц, монитор с разрешением 1366 на 768 точек, частотой 60 Гц и 15,6-и дюймовый.

4.1 Производственная безопасность

Вредным называется производственный фактор, воздействие которого на человека в определенных условиях, приводит к заболеванию или снижению работоспособности. В зависимости от уровня и продолжительности воздействия вредный производственный фактор может стать опасным [28].

Опасным считается производственный фактор, воздействие которого на человека в определенных условиях приводит к травме, а также другому внезапному резкому ухудшению здоровья [31].

Были определены вредные факторы, влияющие на условия труда разработчика, а также произведен анализ источников вредных факторов. Полученные данные представлены в виде сводной таблицы 4.1

Таблица 4.1 - Возможные опасные и вредные факторы

Факторы	Этапы работ	Нормативные
---------	-------------	-------------

(ГОСТ 12.0.003-2015)	Разработка	Исследования	Работа за персональным компьютером	документы
1. Недостаточная освещенность рабочей зоны;		+	+	Параметры освещения определены в СНиП 23-05-95 [27]; Предельно допустимые уровни напряжений прикосновения и токов установлены в ГОСТ 12.1.038-82 [28]; Мероприятия по устранению статического электричества перечислены в ГОСТ 12.1.030-81 [29]; Правила защиты от поражения электрическим током приведены в ГОСТ Р 50571.3-94 [30]; СанПиН 2.2.2.542-96 Гигиенические требования к персональным электронно-вычислительным машинам и организации работ [31].
2. Повышенный уровень статического электричества;		+	+	
3. Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека.	+	+	+	
4. Повышенная или пониженная температура воздуха рабочей зоны	+	+	+	

4.2 Анализ опасных и вредных производственных факторов и обоснование мероприятий по их устранению

4.3.1 Недостаточная освещенность рабочего места

Работа с компьютером подразумевает постоянный зрительный контакт с дисплеем ПЭВМ и занимает от 80 % рабочего времени. Недостаточность освещения снижает производительность труда, увеличивает утомляемость и количество допускаемых ошибок, а также может привести к появлению профессиональных болезней зрения.

Достаточная освещенность рабочего места важна для эффективной работы. Она достигается использованием как естественного, так и искусственного света вместе.

Согласно СНиП 23-05-93 проводимые помещения работы по разработке программного продукта относятся к V разряду зрительной работы. Согласно пункту 4.2 СНиП 23-05-95 [31] уровень общей освещенности должен быть не меньше 200 лк. Величина нормируемой освещенности помещений составляет 300 лк, что является допустимым. Минимальный размер объекта различения 3-5 мм – работа малой точности. Нормы коэффициента пульсации для обозначенного вида работ составляют не более 20%. В помещениях присутствует как естественное, так и искусственное освещение. Последнее обеспечивается люминесцентными источниками света в потолочных светильниках и светильниками, установленными непосредственно на рабочем месте трудящегося.

4.3.2 Повышенный уровень статического электричества

Статическое электричество является одним из опасных факторов. Оно возникает в результате накопления заряда на поверхности или объеме диэлектриков или на изолированных проводниках. Опасность возникает при разряде этого заряда, поскольку заряд может быть велик, что может вызвать поломку оборудования, угрозу для жизни оператора в результате прохождения заряда через тело человека. Накопление заряда может быть вызвано изоляцией оборудования или оператора.

Для устранения необходимо обеспечить заземление компьютера, чтобы заряд мог уходить на землю, не накапливаясь на корпусе компьютера. Правила заземления и другие мероприятия по устранению статического электричества перечислены в ГОСТ 12.1.030-81 [30].

4.3.3 Повышенное значение напряжения в электрической цепи

Нормативное регулирование охраны труда при осуществлении трудовой деятельности за компьютерами осуществляется посредством следующих документов: Типовая инструкция СанПиН 2.2.2.; СанПиН 2.2.2. / 2.4. 1340-03

(далее – СанПиН); ТК РФ; Приказ Минздравсоцразвития РФ № 302н; 426-ФЗ.

Возможность поражения электрическим током на рабочем месте существует, т.к. ЭВМ работает от источника тока. При прикосновении к электроустановкам, по которым протекает ток возможно прохождение тока через тело человека, так как земля имеет нулевой потенциал. Сила поражения зависит от времени воздействия, характера тока, состояния человека, места соприкосновения, пути прохождения тока. Поражающими факторами являются ожоги, сбои в работе сердечной мышцы, нервной системы. Причиной прикосновения является как правило нарушение изоляции изделия и нарушение техники безопасности. Для питания персональных компьютеров в России как правило используется сетевое напряжение 230В переменного тока при частоте 50 Гц. После прохождения через блок питания ток трансформируется в постоянный ток напряжения 5 и 12 В. Согласно ГОСТ 12.1.038-82 [5] допустимо прохождение через тело человека переменного тока с напряжением 2В, силой 0.3мА, постоянного тока 8В, силой 1мА.

Для того чтобы исключить опасность поражения электрическим током необходимо соблюдать правила электрической безопасности:

- 1) Перед включением ЭВМ в сеть должна быть визуально проверена ее электропроводка на отсутствие возможных видимых нарушений изоляции, а также на отсутствие замыкания токопроводящих частей на корпусе компьютера;
- 2) Запрещается при включенной ЭВМ одновременно прикасаться к приборам, имеющим естественное заземление;
- 3) При появлении признаков замыкания необходимо немедленно отключить от электрической сети ЭВМ и устранить неисправность;
- 4) Применения напряжений 42 В и ниже переменного тока частотой 50 Гц и 110 В и ниже постоянного тока;

5) Заземления или зануления корпусов электрооборудования и элементов электроустановок, которые могут оказаться под напряжением вследствие повреждения изоляции.

Для защиты от прохождения тока по телу необходимо соблюдать технику безопасности использования персонального компьютера, использовать ГОСТ Р 50571.3-94 [31] по защите от поражения электрического тока. При обнаружении нарушений изоляции необходимо в первую очередь отключить прибор от источника питания, использовать изоляционные материалы.

4.3.4 Повышенная или пониженная температура воздуха рабочей зоны

Микроклимат помещений – это климат внутренней среды помещений, который определяется действующими на организм сочетаниями температуры, влажности и скорости движения воздуха, а также температуры окружающих поверхностей. Показатели микроклимата должны обеспечивать сохранение теплового баланса человека с окружающей средой и поддержание оптимального или допустимого теплового состояния организма человека [33]. Нормы оптимальных и допустимых метеорологических условий устанавливает СанПиН 2.2.4.548–96. Эти нормы устанавливаются в зависимости от времени года, характера трудового процесса и характера производственного помещения.

Анализируя состояние рабочей зоны, микроклимат которой поддерживается на оптимальном уровне системой водяного центрального отопления и естественной вентиляцией, можно сделать вывод, что параметры микроклимата производственного помещения соответствуют нормам.

4.3 Экологическая безопасность

На сегодняшние дни во всем мире присутствуют проблемы рационального использования природных ресурсов, проблемы охраны окружающей среды. При этом особое внимание уделяется организации воздействия вредных факторов на природу.

С точки зрения потребления ресурсов компьютер потребляет сравнительно небольшое количество электроэнергии, что положительным образом сказывается на общей экономии потребления электроэнергии в целом, соответственно меньше загрязняется окружающая среда.

При выполнении работы на персональных компьютерах, некоторые из частей компьютера могут прийти в негодность или устареть, тогда необходимо их утилизировать. В качестве отходов в результате деятельности программирования может выступать бумага, отходы электроники, ртутьсодержащие лампы. Эти твердые бытовые отходы загрязняют окружающую среду и их необходимо перерабатывать, чтобы максимально снизить их вредное воздействие. Для складирования мусора перед утилизацией предлагается вывоз на территории, предназначенные для этого.

При написании ВКР вредных выбросов в атмосферу, почву и водные источники не производилось, радиационного заражения не произошло, чрезвычайные ситуации не наблюдались, поэтому не оказывались существенные воздействия на окружающую среду.

4.1 Безопасность в чрезвычайных ситуациях

Процесс выполнения работ и написании данная выпускная квалификационная работа, была выполнена в городе Томске, где преобладает континентально-циклонический климат.

Возможными ЧС могут быть сильные морозы, диверсия и пожар. Для Томска в зимнее время года характерны морозы. Достижение критически низких температур приведет к авариям систем теплоснабжения и жизнеобеспечения, приостановке работы и т.д. В случае переморозки труб должны быть предусмотрены запасные обогреватели. Их количества и мощности должно хватать для того, чтобы работа на производстве не прекратилась. Чтобы избежать потерь при авариях на электросетях у каждого компьютера должен быть блок бесперебойного питания, а также должен быть аварийный генератор, который обеспечит дальнейшую бесперебойную работу персонала.

В процессе работы на персональном компьютере возможной чрезвычайной ситуацией, которую создаёт компьютер может стать пожар. Пожар в течении работы персонального компьютера может возникнуть в результате замыкания электропроводки, приводящее к возгоранию аппаратуры. Далее огонь может распространиться на другие объекты рабочего помещения, создавая техногенную чрезвычайную ситуацию.

Федеральный закон №123 "Технический регламент о требованиях пожарной безопасности" [27] устанавливает основные принципы установления пожарной безопасности. Для предупреждения возникновения пожаров необходимо прежде всего следовать технике безопасности пользования персональным компьютером.

Исключение условий образования горючей среды в сложившихся рабочих условиях включает в себя: применение минимального количества горючих веществ и материалов и наиболее безопасного их размещения, изоляция источников зажигания от горючих веществ, понижение концентрации окислителя в горючей среде в защищаемом объеме, удаление из помещений пожароопасных отходов производства, отложений пыли и пуха.

Для предупреждения вероятности осуществления диверсии предприятие необходимо оборудовать системой видеонаблюдения, круглосуточной охраной, пропускной системой, надежной системой связи, а также исключения распространения информации о системе охраны объекта, расположении помещений и оборудования в помещениях, системах охраны, сигнализаторах, их местах установки и количестве.

При возникновении пожара необходимо обеспечить технические и организационные мероприятия по оповещению, эвакуации людей, тушению 75 пожара. При обнаружении пожара необходимо оповестить пожарную службу посредством пожарной сигнализации и провести эвакуацию через эвакуационные пути и выходы. Ликвидация последствий пожара обеспечивается автоматическими установками пожаротушения и пожарными службами.

Выводы и рекомендации по разделу

Проанализировав условия труда на рабочем месте, где была разработана бакалаврская работа, можно сделать вывод, что помещение удовлетворяет необходимым нормам и в случае соблюдения техники безопасности и правил пользования компьютером работа в данном помещении не приведет к ухудшению здоровья пользователя.

Помещение, в котором находится рабочее место, относится к помещениям без повышенной опасности поражения электрическим током. Температурный режим, влажность воздуха не способствуют разрушению изоляции электрооборудования. По состоянию пожаробезопасности помещение соответствует нормам.

Монитор компьютера служит источником электромагнитное поле – вредного фактора, который отрицательно влияет на зрение и в целом на здоровье работника при непрерывной работе более 4 часов. Рекомендуется: во избежание негативного влияния на здоровье необходимо делать перерывы

при работе с ЭВМ и проводить специализированные комплексы упражнений для глаз. При рассмотрении вопроса об охране окружающей среды можно делать вывод, что деятельность помещения не является экологически опасной. По состоянию пожаробезопасности помещение соответствует нормам.

Заключение

В результате выполненного проекта был проведен сбор исходных данных, изучена литература по теме исследования, рассмотрены различные виды алгоритмов и осуществлено их применение на практике.

Проведенный анализ программных инструментов для работы с алгоритмами компьютерного зрения позволил сделать выбор в пользу библиотек OpenCV и Keras. Были рассмотрены существующие наборы COCOdataset (COCO) и Single Shot Detector (SSD), принято решение использовать CIFAR-10 в качестве обучающей выборки, для работы с глубокими нейронными сетями был выбран фреймворк Cafe, по причине её более высокой производительности и обучаемости.

С использованием библиотек OpenCV, NumPY, MatPlotLib была проведена предварительная обработка изображений (преобразование уровня яркости и повышение контрастности для выделения объектов на темных изображениях) для увеличения точности обнаружения объектов.

Построенная и обученная сверточная нейронная сеть обеспечивает удовлетворительную точность распознавания объектов и частичную устойчивость при распознавании объектов, к которым применены операции изменения масштаба, смещения, повороты, смена ракурса и прочие.

ЛИСТИНГ 1

```

# python реализация алгоритма Кэнни //auto_canny.py --imagesimages

# импортировать необходимые пакеты
import numpy as np
import argparse
import glob
import cv2

def auto_canny(image, sigma=0.33):
    # вычислить медиану интенсивности пикселей одного канала
    v = np.median(image)

    # применить автоматическое обнаружение краев Кэнни с использованием
    # вычисленной медианы
    lower = int(max(0, (1.0 - sigma) * v))
    upper = int(min(255, (1.0 + sigma) * v))
    edged = cv2.Canny(image, lower, upper)

    # вернуть изображение (обнаруженные края)
    return edged

# построить аргумент parse и разобрать аргументы
# (argparse.ArgumentParser()-взаимодействовать с системными файлами)
ap = argparse.ArgumentParser()
ap.add_argument("-i", "--images", required=True,
                help="path to input dataset of images")
args = vars(ap.parse_args())

# цикл по изображениям
for imagePath in glob.glob(args["images"] + "/*.jpg"):
    # загрузить изображение, преобразовать его в оттенки серого и размываем
    # по Гауссу,
    # чтобы помочь удалить высокочастотный шум
    image = cv2.imread(imagePath)
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    blurred = cv2.GaussianBlur(gray, (3, 3), 0)

    # применим обнаружение краев Кэнни, используя
    # широкий порог, жесткий порог и автоматически определяемый порог
    wide = cv2.Canny(blurred, 10, 200)
    tight = cv2.Canny(blurred, 225, 250)

```

```
auto = auto_canny(blurred)
```

```
# показываем результата изображение
```

```
cv2.imshow("Original", image)
```

```
cv2.imshow("Edges", np.hstack([wide, tight, auto]))
```

```
cv2.waitKey(0)
```

ЛИСТИНГ 2

```

# импортируем необходимые библиотеки
import numpy as np
import argparse
import cv2

# построить argparse и разобрать аргументы (argparse-
# взаимодействует с командной стр.)
ap = argparse.ArgumentParser()
ap.add_argument("-i", "--image", required=True,
help="path to input image")
ap.add_argument("-p", "--prototxt", required=True,
help="path to Caffe 'deploy' prototxt file")
ap.add_argument("-m", "--model", required=True,
help="path to Caffe pre-trained model")
ap.add_argument("-c", "--confidence", type=float, default=0.2,
help="minimum probability to filter weak detections")
args = vars(ap.parse_args())

# инициализировать список меток классов обнаруженных объектов,
# а затем сгенерировать набор цветов ограничивающего прямоугольника для
каждого класса
CLASSES = ["background", "aeroplane", "bicycle", "bird", "boat",
"bottle", "bus", "car", "cat", "chair", "cow", "diningtable",
"dog", "horse", "motorbike", "person", "pottedplant", "sheep",
"sofa", "train", "tvmonitor"]
COLORS = np.random.uniform(0, 255, size=(len(CLASSES), 3))

# загружаем модель из диска
print("[INFO] loading model...")
net = cv2.dnn.readNetFromCaffe(args["prototxt"], args["model"])

# подготовим наш объект, который мы передадим через сеть:
# загружаем входное изображение и делаем входной объект для
изображения,
# изменив размер до фиксированных 300x300 пикселей
image = cv2.imread(args["image"])
(h, w) = image.shape[:2]
blob = cv2.dnn.blobFromImage(cv2.resize(image, (300, 300)), 0.007843, (300,
300), 127.5)

# передать blob (BLOB-это просто изображений с одинаковыми
пространственными размерами,

```

```

# которые должны быть предварительно обработаны одинаковым образом.)
# через сеть и получить обнаружения и прогнозы
print("[INFO] computing object detections...")
net.setInput(blob)
detections = net.forward()

# Цикл на обнаружения объектов
for i in np.arange(0, detections.shape[2]):
    # извлекать вероятность, связанную с прогнозом
    confidence = detections[0, 0, i, 2]

    # отфильтровывать слабые обнаружения,
    # гарантируя, что «точность» обнаруженного объекта превышает
    # минимальную нам заданную точность.
    if confidence > args["confidence"]:
        # извлечь индекс метки класса из `detections`,
        # а затем вычислить (x, y)-координаты выделенной рамки для объекта
        idx = int(detections[0, 0, i, 1])
        box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
        (startX, startY, endX, endY) = box.astype("int")

        # отобразить точность обнаруженного объекта
        label = "{}: {:.2f}%".format(CLASSES[idx], confidence * 100)
        print("[INFO] {}".format(label))
        cv2.rectangle(image, (startX, startY), (endX, endY),
            COLORS[idx], 2)
        y = startY - 15 if startY - 15 > 15 else startY + 15
        cv2.putText(image, label, (startX, y),
            cv2.FONT_HERSHEY_SIMPLEX, 0.5, COLORS[idx], 2)

# показывать выходное изображение
cv2.imshow("Output", image)
cv2.waitKey(0)

```

Список литературы

1. А. Джулли, С. Пал «Библиотека Keras – инструмент глубокого обучения». / пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2018. – 294 с.
2. Я. Э. Солем «Программирование компьютерного зрения на языке Python». / пер. с англ. Слинкин А. А. 2016– 332 с.
3. Веб-сервис и его архитектура // [Электронный ресурс]. – Режим доступа: <https://www.guru99.com/web-service-architecture.html>
4. Компоненты веб-сервисов// [Электронный ресурс]. – Режим доступа: https://www.tutorialspoint.com/webservices/what_are_web_services.htm
5. Введение HTML// [Электронный ресурс]. – Режим доступа: <https://www.geeksforgeeks.org/html-introduction/>
6. Р. Н. Дженнифер «HTML5, CSS3 и JavaScript. Исчерпывающее руководство». / пер. с англ. М. А. Райтман. (2018), 288 с.
7. Основа JavaScript // [Электронный ресурс]. – Режим доступа: <https://www.makeuseof.com/tag/what-is-javascript/>
8. А.А. Головатый, Дж. Каплан-Мосс«Django. Подробное руководство»./2-е издание. – Пер. с англ. – СПб.: СимволПлюс, (2010)
9. А. Rosebrock «Deep Learning for Computer Vision with Python». English. (2017)
10. Как работает компьютерное зрение// [Электронный ресурс]. – Режим доступа: <https://blog.algorithmia.com/introduction-to-computer-vision/>
11. Tsz-Ho Yu, Yui-Sang Moon: Unsupervised abnormal behavior detection for real-time surveillance using observed history // [Электронный ресурс]. – Режим доступа: <https://pdfs.semanticscholar.org/dacd/5ae4d3ffd237281b7559f011ee59563d5fae.pdf>

12. Что такое свёрточная сеть // [Электронный ресурс]. – Режим доступа: <https://habr.com/post/309508/>
13. Zhe Cao, Tomas Simon, Shih-En Wei Yaser, Sheikh The Robotics Institute, Carnegie Mellon, University: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields // [Электронный ресурс]. – Режим доступа: <https://arxiv.org/pdf/1611.08050.pdf>
14. Цветовая модель RGB // [Электронный ресурс]. – Режим доступа: http://delovoygorod.com/e_rgb.html
15. А. Джулли, С. Пал «Библиотека Keras – инструмент глубокого обучения». / пер. с англ. Слинкин А. А. – М.: ДМК Пресс, (2018).
16. Сверточная нейронная сеть // [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/348000/>
17. Пару слов о распознавании объектов // [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/208090/>
18. Обнаружение границ Кэнни шаг за шагом в Python - компьютерное видение// [Электронный ресурс]. – Режим доступа: <https://towardsdatascience.com/canny-edge-detection-step-by-step-in-python-computer-vision-b49c3a2d8123>
19. OpenCV: Detecting Edges, Lines, and Shapes // [Электронный ресурс]. – Режим доступа: <https://hub.packtpub.com/opencv-detecting-edges-lines-shapes/>
20. Николенко С., Кадури А., Архангельская Е. «Глубокое обучение». — СПб.: Питер, 2018. — 480 с.
21. Нулевой параметр, автоматическое определение краев Canny с Python и OpenCV // [Электронный ресурс]. – Режим доступа: <https://www.pyimagesearch.com/2015/04/06/zero-parameter-automatic-canny-edge-detection-with-python-and-opencv/>
22. Официальный ресурс библиотеки OpenCV // [Электронный ресурс]. – Режим доступа: <https://opencv.org/>

23. Официальный ресурс фреймворкаCaffe // [Электронный ресурс]. – Режим доступа: <http://caffe.berkeleyvision.org/>
24. Официальный ресурс фреймворкаTensorFlow // [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org/>
25. Форматы изображений // [Электронный ресурс]. – Режим доступа:<http://blogmonet.ru/chem-otlichayutsya-formaty-izobrazhenij-bmp-png-jpg-gif.html>
26. Набор данных CIFAR-10// [Электронный ресурс]. – Режим доступа:<https://www.cs.toronto.edu/~kriz/cifar.html>
27. Deep Learning for Computer Vision – Introduction to Convolution Neural Networks // [Электронный ресурс]. – <https://www.analyticsvidhya.com/blog/2016/04/deep-learning-computer-vision-introduction-convolution-neural-networks/> Дата обращения 10.06.2019.
28. Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 27.12.2018)
29. ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования.
30. ГОСТ 12.2.033-78 ССБТ. Рабочее место при выполнении работ стоя. Общие эргономические требования.
31. ГОСТ 12.2.049-80 ССБТ. Оборудование производственное. Общие эргономические требования.
32. ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление, зануление.
33. ГОСТ 12.1.019-2017 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты.
34. СанПиН 2.2.2.542-96 Гигиенические требования к видеодисплейным терминалам, персональным электронно-вычислительным машинам и организации работ. Режим доступа: <http://docs.cntd.ru/document/5200235>

35. Безопасность жизнедеятельности: Учебник для вузов/ С.В. Белов, А.В. Ильницкая, А.Ф. Козьяков и др.; Под общ. ред. С.В. Белова. 7-е изд., стер. – М.: Высш.шк., 2007. – 616 с.