

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное
учреждение высшего образования
«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

На правах рукописи



Ковалев Владислав Владимирович

**МЕТОД И АЛГОРИТМЫ РАСПОЗНАВАНИЯ
МАЛОРАЗМЕРНЫХ ИЗОБРАЖЕНИЙ ПОДВИЖНЫХ
ОБЪЕКТОВ НА УСТРОЙСТВАХ С ОГРАНИЧЕННЫМ
ВЫЧИСЛИТЕЛЬНЫМ РЕСУРСОМ**

Специальность — 1.2.1. Искусственный интеллект и машинное обучение
(технические науки)

Диссертация на соискание учёной степени
кандидата технических наук

Научный руководитель:
доктор технических наук, доцент
Сергеев Николай Евгеньевич

Таганрог — 2025

Оглавление

	Стр.
Введение	6
Глава 1. АНАЛИЗ ПРОБЛЕМЫ РАСПОЗНАВАНИЯ МАЛОРАЗМЕРНЫХ ИЗОБРАЖЕНИЙ	19
1.1 Основные понятия распознавания образов	19
1.2 Классификация задач распознавания образов	20
1.3 Критерий малоразмерности изображений	22
1.4 Критерии качества в задаче распознавания	25
1.5 Методы и алгоритмы решения задачи распознавания изображений .	28
1.6 Переобучение свёрточных нейронных сетей в задаче распознавания малоразмерных изображений	33
1.7 Регуляризация свёрточных нейронных сетей	37
1.8 Датасеты для задачи распознавания изображений	43
1.9 Построение свёрточных нейронных сетей на устройствах с ограниченным вычислительным ресурсом	45
1.10 Выбор вычислительного устройства для построения свёрточных нейронных сетей на устройствах с ограниченным вычислительным ресурсом	47
1.11 Выводы по материалам первой главы	51
Глава 2. РАСПОЗНАВАНИЕ МАЛОРАЗМЕРНЫХ ИЗОБРАЖЕНИЙ . .	53
2.1 Разработка метода расширения признакового пространства с целью повышения информативности системы признаков	53
2.2 Постановка задачи	54
2.3 Метод расширения признакового пространства с целью повышения информативности системы признаков	56
2.4 Обучение и распознавание с применением разработанного метода расширения признакового пространства	60

	Стр.
2.5 Распознавание малоразмерных изображений свёрточными нейронными сетями	62
2.6 Разработка алгоритма предварительной обработки изображений с целью комплексирования признаков движения	66
2.7 Разработка алгоритма формирования аннотированных условно-реальных разномодальных изображений	70
2.8 Формирование комплексированных изображений	75
2.9 Выводы по материалам второй главы	77
Глава 3. РАЗРАБОТКА АЛГОРИТМОВ ПОСТРОЕНИЯ СВЁРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ НА УСТРОЙСТВАХ С ОГРАНИЧЕННЫМ ВЫЧИСЛИТЕЛЬНЫМ РЕСУРСОМ	78
3.1 Разработка структурной схемы алгоритма распознавания изображений с применением алгоритма предварительной обработки изображений на нейронном ускорителе	78
3.2 Разработка алгоритма предварительной обработки изображений с целью комплексирования признаков движения на нейронном ускорителе	80
3.3 Разработка алгоритма построения системы распознавания малоразмерных изображений на устройствах с ограниченным вычислительным ресурсом	82
3.4 Выводы по материалам третьей главы	90
Глава 4. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ РАЗРАБОТАННЫХ МЕТОДА И АЛГОРИТМОВ	91

4.1	Вычислительный эксперимент с целью исследования метода расширения признакового пространства и алгоритма предварительной обработки изображений	91
4.1.1	Описание вычислительного эксперимента с целью исследования метода расширения признакового пространства и алгоритма предварительной обработки изображений	91
4.1.2	Формирование датасета для задачи распознавания малоразмерных изображений объектов	92
4.1.3	Обучение тестовых архитектур свёрточных нейронных сетей	94
4.1.4	Оценка информативности системы признаков	100
4.1.5	Оценка качества распознавания малоразмерных изображений объектов свёрточных нейронных сетей	101
4.2	Вычислительный эксперимент с целью исследования разработанного алгоритма предварительной обработки изображений на нейронном ускорителе	109
4.2.1	Описание вычислительного эксперимента с целью исследования алгоритма предварительной обработки изображений на нейронном ускорителе	109
4.2.2	Вычислительные устройства с нейронным ускорителем	112
4.2.3	Оценка времени выполнения разработанного алгоритма предварительной обработки изображений на нейронном ускорителе	115
4.2.4	Оценка качества распознавания и времени выполнения свёрточных нейронных сетей при переходе к вычислениям с пониженной разрядностью	117

4.3	Разработка интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом	119
4.3.1	Описание исходного состава компонент и требуемых характеристик распознающей системы	119
4.3.2	Построение интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом	120
4.4	Выводы по материалам четвёртой главы	124
	Заключение	126
	Список сокращений и условных обозначений	130
	Список литературы	132
	Приложение А. Листинги программ	147
	Приложение Б. Результаты вычислительного эксперимента построения интеллектуальной системы распознавания малоразмерных изображений на устройствах с ограниченным вычислительным ресурсом	158
	Приложение В. Акты о внедрении результатов работы	159

Введение

Актуальность темы исследования и степень её разработанности. Одним из наиболее важных и активно развиваемым в мире технологическим направлением является цифровая обработка изображений. Автоматическое распознавание изображений объектов является востребованной задачей этого направления на текущем научно-техническом этапе. Алгоритмы распознавания находят своё применение в таких областях науки и техники, как: трекинг лиц в мобильных приложениях, охранные системы видеонаблюдения, медицинская диагностика, автоматизация управления беспилотных автомобилей, автомобильная промышленность и др.

Развитие методов и алгоритмов распознавания изображений объектов ведётся уже десятилетия. Применяемые подходы можно условно разделить на классические методы, основанные на априорных знаниях о статистических характеристиках тренировочных выборок, и методы машинного обучения. Свёрточные нейронные сети, которые включают методы машинного обучения, превосходят все современные решения на текущий момент времени. Развитию свёрточных нейронных сетей способствовало увеличение вычислительной мощности аппаратуры, появление больших объёмов тренировочных данных в открытом доступе и новых методов глубокого обучения. Алгоритмы распознавания на базе свёрточных нейронных сетей можно условно разделить на две категории: одноэтапные и двухэтапные подходы. Двухэтапные подходы формируют множество потенциальных регионов на изображении и классифицируют их. Одноэтапные подходы не формируют множество потенциальных регионов на изображении, а напрямую распознают изображения объектов. Двухэтапные подходы обеспечивают большие показатели качества распознавания в отличие от одноэтапных подходов, однако проигрывают в вычислительной сложности. Поэтому разработка и исследование алгоритмов обработки изображений на базе свёрточных нейронных сетей, удовлетворяющих необходимым характеристикам в качестве распознава-

ния и быстроедействие, является актуальной задачей как с теоретической, так и с практической точки зрения.

Обычно задачу распознавания в системах видеоаналитики и видеонаблюдения, беспилотных автомобилях и др. выполняют свёрточные нейронные сети. Такие системы требуют раннего обнаружения и распознавания целей, чтобы обеспечить больше времени на реагирование для снижения риска аварийных ситуаций и обеспечения безопасности человеческих жизней и др. В задаче раннего обнаружения изображения объектов представлены малой группой пикселей, которая претерпевает яркостные и геометрические изменения для каждого момента времени. Размер изображения объекта зависит не только от дальности и физического размера объекта, а также от оптических характеристик видеокамеры и параметров объектива: разрешения матрицы, светочувствительности матрицы, фокусного расстояния объектива и др. Обнаружение и распознавание таких объектов вызывает трудности даже у современных архитектур свёрточных нейронных сетей. Это связано с тем, что свёрточные нейронные сети в процессе обучения иногда не могут найти некоторую взаимосвязь между входными и выходными данными. Такую взаимосвязь называют обобщающей способностью.

Как показывает практика, в задаче распознавания малоразмерных изображений нейросетевой алгоритм не может обобщить данные из-за проблемы переобучения. Переобучение возникает, когда модель нейронной сети слишком сложна относительно объёма признакового пространства. Предотвращение переобучения свёрточных нейронных сетей возможно за счет расширения признакового пространства новыми независимыми признаками движения. Чтобы идентифицировать признак движения динамических объектов, необходимо обрабатывать упорядоченную во времени последовательность кадров, которая содержит признаки объектов в процессе их изменения. Идентифицировать наличие движения по видеопоследовательности можно классическими подходами технического зрения: статистическими методами, разностными методами, методами вычисления оптического потока и др. Задачу идентификации движения в видеопоследовательности можно разбить на случаи со статичным и подвижным задним

фоном. Идентификация движения с подвижным фоном является более сложной задачей, поскольку приходится учитывать межкадровое смещение, перспективные искажения и другие геометрические изменения изображений.

Как правило, системы анализа и обработки видеопоследовательности предназначены для установки на автономных носителях [1]. Такие системы принимают и обрабатывают большие объёмы данных с различных сенсорных датчиков в различных спектральных диапазонах. К числу целевых объектов, подлежащих распознаванию, относятся легковые автомобили, грузовые автомобили, трактора и др. Обеспечение необходимого времени реагирования в системах раннего распознавания напрямую зависит от времени обработки данных, которое коррелирует с производительностью вычислительного устройства и сложностью алгоритма. Решение этой проблемы вызывает интерес к исследованию и разработке методов, которые позволят реализовывать нейросетевые алгоритмы цифровой обработки данных на встраиваемых вычислительных устройствах в масштабе реального времени.

В силу всего вышесказанного, разработка и исследование новых методов и алгоритмов на базе искусственного интеллекта и машинного обучения, решающих задачу распознавания малоразмерных изображений объектов, является актуальной. Создание системы искусственного интеллекта для обработки видеоданных, обеспечивающей требуемые характеристики в качестве распознавания, скорости обработки данных и потребляемой мощности вычислительной аппаратуры, носит практически ценный характер.

Большое количество работ, посвященных проблеме распознавания малоразмерных изображений объектов, говорит об актуальности исследования данного направления. Первые алгоритмы распознавания изображений объектов, в том числе и малоразмерных изображений, основаны на построение усиленных решающих функций в виде линейных комбинаций слабых решающих функций. К числу таких алгоритмов можно отнести каскадный детектор изображений Виолы-Джонса, представленный авторами P. Viola и M. Jones [2, 3] в 2001 году, алгоритм Histogram of Oriented Gradients (HOG), предложенный авторами N. Dalal и B.

Triggs [4] в 2005 году и алгоритм Deformable Part-based Model (DPM), предложенный авторами P. Felzenszwalb [5] и др.

В 2016 году авторы Chen C., Liu M.-Y., Tuzel O., Xiao J. опубликовали одну из первых работ [6], посвященную распознаванию малоразмерных изображений объектов. В качестве алгоритма распознавания используется двухэтапная свёрточная нейронная сеть R-CNN [7]. Повышение качества распознавания нейросетевого алгоритма достигается за счет аугментации данных, применения контекстной информации и модификации архитектуры свёрточной нейронной сети. Последующие работы [8–13], посвящённые распознаванию малоразмерных изображений объектов, используют подходы, в основе которых лежит парадигма подходов, рассмотренных в статье [6]. В большинстве случаев двухэтапные свёрточные нейронные сети типа R-CNN не применяются в автономных бортовых системах из-за высокой сложности алгоритма. Поэтому применяют одноэтапные архитектуры свёрточных нейронных сетей, среди которых можно выделить зарекомендованные архитектуры: YOLO V3 [14], SSD [15], CenterNet [16] и др.

В 2022 году авторы Akyon F. C., Onur Altinuc S., Temizel A. предложили алгоритм Slicing Aided Hyper Inference (SAHI) [17], который позволяет повысить качество распознавания малоразмерных изображений объектов. Алгоритм основан на базе одноэтапной свёрточной нейронной сети FCOS [18], который разбивает входное изображение на количество n подизображений с перекрытием. Каждое подизображение анализируется на наличие изображений целевых объектов.

Повышение качества распознавания обеспечивается за счет аугментации данных и тонкой настройки (fine-tuning) модели свёрточной нейронной сети. Алгоритм SAHI подразумевает последовательную обработку изображений, что требует меньше ресурса памяти вычислительной аппаратуры по сравнению с обработкой всего изображения. Но с другой стороны, увеличивается время обработки данных алгоритмом. Применение SAHI целесообразно в обработке крупномасштабных изображений, где нет существенных ограничений, связанных со

временем выполнения. В бортовых системах раннего распознавания применение такого алгоритма не целесообразно.

Другим способом повышения качества распознавания свёрточными нейронными сетями является анализ последовательности видеок кадров рекурсивными алгоритмами Long Short-Term Memory (LSTM). С одной стороны, применение LSTM учитывает признаки, характеризующие временную зависимость, но с другой – применение LSTM увеличивает вычислительную сложность нейросетевого алгоритма, обладает более сложной процедурой обучения, и порой может выдавать непредсказуемый результат из-за своей рекурсивной природы.

Расширение исходного признакового пространства данными с лазерных дальномеров, радиолокационными изображениями, инфракрасными изображениями и видеок кадрами с признаками движения является зарекомендованным способом повышения качества распознавания малоразмерных изображений алгоритмами искусственного интеллекта и машинного обучения [19–23]. Однако только видеок кадры с признаками движения можно получить без усложнения распознающей системы добавлением различных сенсорных датчиков. Существующие алгоритмы не учитывают признаки движения в процессе продолжительного наблюдения за объектом. В результате получается парадоксальная ситуация, когда современные алгоритмы на базе искусственного интеллекта не используют весь объём имеющихся признаков, содержащихся в последовательности видеок кадров, для повышения качества распознавания малоразмерных изображений объектов свёрточными нейронными сетями.

В данной работе предложены новые научно-технологические разработки, позволяющие решить научно-техническую задачу, состоящую в создании интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом, что имеет существенное значение для развития области «Технологии создания высокоскоростных транспортных средств и интеллектуальных систем управления новыми видами транспорта» [24] из Перечня критических технологий Российской Федерации и разработок в сфере искусственного интеллекта и машинного обучения.

Объектом исследования являются малоразмерные изображения объектов.

Предметом исследования являются методы и алгоритмы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом.

Целью работы является повышение качества распознавания малоразмерных изображений объектов свёрточными нейронными сетями на устройствах с ограниченным вычислительным ресурсом.

Для решения общей научной задачи в диссертационной работе необходимо решить следующие частные **основные задачи**:

1. Провести обзор и анализ перспективных методов и алгоритмов распознавания малоразмерных изображений.
2. Разработать метод расширения признакового пространства, который позволит повысить информативность системы признаков изображений для задачи распознавания малоразмерных изображений свёрточными нейронными сетями.
3. Разработать алгоритм предварительной обработки изображений с целью комплексирования признаков движения.
4. Сформировать датасеты разномодальных изображений для задачи распознавания малоразмерных изображений целевых объектов.
5. Провести вычислительный эксперимент с целью исследования разномодальных изображений на информативность системы признаков.
6. Провести вычислительный эксперимент с целью исследования влияния разработанного алгоритма предварительной обработки изображений на качество распознавания малоразмерных изображений свёрточных нейронных сетей.
7. Разработать алгоритм предварительной обработки изображений с целью комплексирования признаков движения для нейронного ускорителя.
8. Провести вычислительный эксперимент с целью исследования временных характеристик алгоритма предварительной обработки изображений на нейронных ускорителях.

9. Разработать алгоритм создания интеллектуальной системы распознавания малоразмерных изображений на устройствах с ограниченным вычислительным ресурсом.
10. Построить интеллектуальную систему распознавания малоразмерных изображений на устройствах с ограниченным вычислительным ресурсом.

Методология и методы диссертационного исследования. Для решения выявленных проблем в диссертационной работе использовались теория и методы машинного обучения, распознавания образов, алгоритмы цифровой обработки изображений, методы и алгоритмы предварительной обработки данных, методы статистического анализа, а также разные технологии программирования.

Научная новизна полученных результатов. В рамках данного исследования получены новые научные результаты:

1. Разработан метод расширения признакового пространства, повышающий информативность системы признаков, что позволило снизить частоту ошибок распознавания малоразмерных изображений объектов на 11.5 – 14.5 % (п. 4 паспорта специальности; страницы 53-62, 91-100 диссертации).
2. Разработан алгоритм предварительной обработки изображений, который учитывает больше признаков движения, что позволило повысить вероятность распознавания малоразмерных изображений объектов свёрточными нейронными сетями на 14.9 – 20.1 % (п. 4 паспорта специальности; страницы 66-70, 101-108 диссертации).
3. Разработан алгоритм формирования условно-реальных данных, отличающийся возможностью генерировать аннотированные разномодальные изображения, позволяющий заменить ручной процесс разметки данных автоматическим (п. 14 паспорта специальности; страницы 70-75, 92-94 диссертации).
4. Разработан алгоритм предварительной обработки изображений с целью комплексирования признаков движения на нейронном ускорителе, что

позволило сократить время выполнения алгоритма в 1.3 – 3.2 раза (п. 4 паспорта специальности; страницы 78-82, 109-119 диссертации).

5. Создана интеллектуальная система распознавания малоразмерных изображений целевых объектов, которая обеспечивает требуемые характеристики в качестве распознавания, скорости обработки данных и допустимой потребляемой мощности вычислительного устройства (п. 4 паспорта специальности; страницы 82-89, 119-123 диссертации).

Новые результаты, полученные в ходе выполнения диссертационного исследования, соответствуют паспорту научной специальности 1.2.1. Искусственный интеллект и машинное обучение (технические науки) в части следующих пунктов:

П. 4. Разработка методов, алгоритмов и создание систем искусственного интеллекта и машинного обучения для обработки и анализа текстов на естественном языке, для изображений, речи, биомедицины и других специальных видов данных.

П. 14. Методы и средства формирования массивов условно-реальных данных и прецедентов, необходимых для решения задач искусственного интеллекта и машинного обучения.

Практическая значимость заключается в возможности использования разработанных алгоритмических и программных решений для создания интеллектуальных систем распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом, удовлетворяющих заданным характеристикам в качестве распознавания, скорости обработки данных и допустимой потребляемой мощности.

Теоретическая значимость заключается в возможности применения подходов увеличения информативности системы признаков, рассмотренных в разработанном методе расширения признакового пространства, для усовершенствования методов распознавания образов.

Положения, выносимые на защиту:

1. Метод расширения исходного признакового пространства, который позволяет повысить информативность системы признаков за счёт добавления признаков движения.
2. Алгоритм предварительной обработки изображений с целью комплексования признаков движения на центральном процессоре, который позволяет повысить качество распознавания малоразмерных изображений свёрточными нейронными сетями.
3. Алгоритм формирования условно-реальных данных, который позволяет генерировать в автоматическом режиме аннотированные разномодальные изображения.
4. Алгоритм предварительной обработки изображений с целью комплексования признаков движения на нейронном ускорителе, который позволяет повысить скорость обработки данных.
5. Алгоритм построения интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом, которая удовлетворяет требуемым характеристикам.

Достоверность полученных результатов. Достоверность полученных результатов диссертационной работы подтверждена корректным проведением вычислительных экспериментов, основанных на валидных данных, репрезентабельных критериях и не противоречит выдвинутым теоретическим предположениям.

Реализация и внедрение результатов работы. Основные результаты диссертационного исследования внедрены в учебный процесс кафедры вычислительной техники ФГАОУ ВО «Южный федеральный университет» при подготовке бакалавров основной образовательной программы «Технологии искусственного интеллекта» по направлению 09.03.01 «Информатика и вычислительная техника» и магистров основной образовательной программы «Высокопроизводительные вычислительные системы и квантовая обработ-

ка информации» 09.04.01 по направлению «Информатика и вычислительная техника» (см. Приложение В). Теоретические и практические результаты диссертационной работы внедрены в научно-исследовательской работе «Гепард-Деталь-НКБ ВСБЦВМ-ЭИИ», выполненной в «Акционерное общество Научно-конструкторское бюро вычислительных систем» (см. Приложение В).

Апробация работы. Основные положения и результаты диссертационной работы докладывались и обсуждались: на XVIII Всероссийской научной конференции молодых учёных, аспирантов и студентов «Информационные технологии, системный анализ и управление» (Таганрог, 2020); VII, IX Всероссийской научно-технической конференции «Фундаментальные и прикладные аспекты компьютерных технологий и информационной безопасности» (Таганрог, 2021, 2023); Международном научно-техническом конгрессе «Интеллектуальные системы и информационные технологии – 2023».

Публикации. Основные результаты по теме диссертации изложены в 13 печатных изданиях, 6 из которых опубликованы в журналах, рекомендованных ВАК РФ, 7 статей в источниках, индексируемых в РИНЦ.

Противоречие в практике состоит в том, что разработка алгоритмов распознавания малоразмерных изображений для устройств с ограниченным вычислительным ресурсом задаёт высокие требования к качеству распознавания и скорости обработки данных. С одной стороны, высокое качество распознавания можно обеспечить вычислительно сложными свёрточными нейронными сетями. С другой стороны, распознавание малоразмерных изображений свёрточными нейронными сетями на устройствах с ограниченным вычислительным ресурсом не позволяет обрабатывать данные с необходимой скоростью.

Противоречие в теории обусловлено тем, что свёрточные нейронные сети в процессе обучения могут начать переобучаться, увеличивая вероятность ошибок классификации на тестовых данных. Неоправданное расширение признакового пространства новыми признаками не гарантирует предотвращение переобучения из-за недостаточной различительности новых признаков или даже может ускорить переобучение.

Объём и структура работы. Диссертация состоит из введения, 4 глав, заключения и 3 приложений. Полный объём диссертации составляет 162 страницы, включая 55 рисунков и 17 таблиц. Список литературы содержит 139 наименований.

В первой главе «Анализ проблемы распознавания малоразмерных изображений» выявлены проблемы, связанные с высокими затратами на разработку интеллектуальной системы распознавания малоразмерных изображений объектов, которые обусловлены высокой стоимостью процедуры распознавания и стоимостью потерь распознавания. В результате анализа причин, определяющих высокие затраты разработки распознающей системы, выявлен ряд существенных проблем, связанных с неудовлетворительным качеством распознавания, высокой стоимостью разработки процедуры распознавания, неприемлемой временной задержкой обработки данных и сложностью выбора необходимого вычислительного устройства.

Во второй главе «Расознавание малоразмерных изображений» рассмотрено решение первых трёх проблем диссертационной работы. Разработан метод расширения признакового пространства с целью повышения информативности системы признаков. Для решения проблемы неудовлетворительного качества распознавания малоразмерных изображений свёрточными нейронными сетями предложен алгоритм предварительной обработки изображений с целью комплексирования признаков движения, который формирует комплексированное изображение с большим количеством различительных признаков. Свёрточные нейронные сети, обученные на комплексированных изображениях, улучшили качество распознавания малоразмерных изображений. Разработан алгоритм автоматического формирования условно-реальных аннотированных разномодальных изображений, который позволяет уменьшить стоимость разработки распознающей системы за счёт снижения затрат на формирование данных.

В третьей главе «Разработка алгоритмов построения свёрточных нейронных сетей на устройствах с ограниченным вычислительным ресурсом» рассмотрено решение четвертой и пятой проблем. Для решения четвёртой про-

блемы разработан алгоритм предварительной обработки изображений с целью комплексирования признаков движения на нейронном ускорителе. Благодаря объединению алгоритма предварительной обработки изображений с экстрактором признаков свёрточной нейронной сети и выполнению на нейронном ускорителе обеспечивается повышение скорости обработки данных.

Разработан алгоритм построения интеллектуальной системы распознавания малоразмерных изображений объектов, с помощью которого можно создать интеллектуальную распознающую систему, удовлетворяющую требуемым качеству распознавания, скорости обработки данных и допустимой потребляемой мощности вычислительного устройства.

В четвёртой главе «Экспериментальное исследование разработанных метода и алгоритмов» проведена апробация разработанного метода расширения признакового пространства и алгоритма предварительной обработки изображений с целью комплексирования признаков движения путём проведения вычислительного эксперимента.

Сформировано множество тестовых архитектур свёрточных нейронных сетей с различной вычислительной сложностью. Сгенерированы тренировочный и тестовый датасеты с помощью разработанного алгоритма формирования аннотированных условно-реальных разномодальных изображений. Проведено обучение тестовых моделей свёрточных нейронных сетей на датасетах.

Проведены вычислительные эксперименты с целью исследования влияния разработанного метода расширения признакового пространства на информативность системы признаков и алгоритма предварительной обработки изображений на качество распознавания малоразмерных изображений объектов свёрточными нейронными сетями. Полученные результаты вычислительных экспериментов подтверждают эффективность разработанного метода и алгоритма.

Разработана интеллектуальная система распознавания малоразмерных изображений объектов, которая обеспечивает требуемые характеристики в качестве распознавания, скорости обработки данных и допустимой потребляемой мощности вычислительного устройства. Система создана на основе

набора, состоящего: из вычислительного устройства NVIDIA Jetson AGX Xavier, фреймворка TensorRT, архитектуры свёрточной нейронной сети YOLOv5s-M, половинной разрядности вычислений и датасетов комплексированных изображений с двумя разностными каналами. Набор получен с помощью разработанного алгоритма построения системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом.

Глава 1. АНАЛИЗ ПРОБЛЕМЫ РАСПОЗНАВАНИЯ МАЛОРАЗМЕРНЫХ ИЗОБРАЖЕНИЙ

1.1 Основные понятия распознавания образов

Спектр задач, решаемых распознающими системами, чрезвычайно широк. На таких системах реализуются алгоритмы автоматического распознавания изображений объектов. В основе современных алгоритмов лежит концепция методов глубокого обучения. Прежде чем проводить процедуру распознавания объектов реального мира распознающей системой, необходимо сформировать информацию об объектах. Распознающая система собирает информацию об объектах с помощью сенсорных датчиков. Отражение какого-либо объекта на «органы восприятия» распознающей системы называется *изображением объекта*. Изображение объекта – это конкретная характеристика объекта. Количественное описание характерных свойств изображения объекта называется *набором признаков*, который приближённо описывает интересующий объект реального мира. Множество изображений объектов, объединённых некоторыми общими признаками, называется *образом* [1, 25].

В распознавании визуальных образов сенсорные датчики распознающих систем фиксируют электромагнитное излучение оптического диапазона. К оптическому диапазону относятся те виды электромагнитного излучения, которые подчиняются законам геометрической оптики, а также свет очень слабо взаимодействует с веществом [26]. Оптический диапазон состоит из следующих видов излучения: рентгеновское, ультрафиолетовое, видимое, инфракрасное (ИК).

Эти типы излучений фиксируются специализированной аппаратурой, в основе аппаратной составляющей этих устройств лежит детектор излучения – матрица. Матрица состоит из множества упорядоченных дискретных элементов, чувствительных к излучению конкретного диапазона. Сенсорное устройство фор-

мирует дискретное изображение в градациях серого, которое можно представить в виде функции двух переменных $F(x,y)$, где x и y – координаты на двумерной плоскости изображения. Единичный элемент дискретного изображения называется пикселем, который характеризуется интенсивностью (яркостью) и имеет 256 оттенков серого (8 бит) от 0 до 255. Временная последовательность дискретных изображений называется видеокадрами, которые можно представить в виде функции трёх переменных $F(x,y,t)$, где t – время. Видеокадры имеют период появления во времени, который зависит от частоты выдачи изображений камерой.

1.2 Классификация задач распознавания образов

Задача распознавания состоит из поиска изображений объектов на изображении с последующей классификацией. Задача поиска заключается в определении локальной области на изображении, в которой расположено изображение целевого объекта. Таким образом, локальная область на изображении оценивается на принадлежность к двум классам: изображение целевого объекта или фона [27]. Под задачей классификации понимается определение принадлежности обнаруженного изображения объекта к одному из взаимоисключающих классов [28, 29] путем анализа набора признаков. Далее рассмотрим формальную постановку задачи распознавания.

В общем случае распознающая система состоит из блока измерения признаков (рецептора X) и блока принятия решения (классификатора D). Рецептор измеряет значения признаков X реализации образа. Классификатор в соответствии с определённой функцией D относит реализации образов к определенным областям S выбранного признакового пространства.

Задача разработки распознающего устройства сводится к минимизации затрат N , состоящих из стоимости процедуры N_1 и стоимости потерь N_2 . Под стоимостью процедуры N_1 понимается мера, с помощью которой можно оценивать такие характеристики как, объём памяти, вычислительную сложность,

вес, габариты, стоимость оборудования и др. В N_1 входят стоимость реализации решающей функции $N(D)$, стоимость измерения признаков $N(X)$, стоимость описания эталонов (формирования разметки) $N(S)$:

$$N_1 = N(D) + N(X) + N(S). \quad (1.1)$$

Под стоимостью потерь N_2 понимается стоимость ошибок распознавания, которые зависят от вероятности этих ошибок. При разработке распознающей системы принято, чтобы затраты N не превышали N_0 .

$$N_1 + N_2 \leq N_0. \quad (1.2)$$

Для случая, когда заданы величина потерь N_0 и два из трёх элементов S , X , D и требуется выбрать третий элемент, вытекает формулировка трёх основных типов задач распознавания [25, 30]:

1. Задано множество образов S , указаны признаки X , по которым реализации образов будут отличаться друг от друга, и величина потерь N_0 . Требуется найти такую решающую функцию D , которая будучи максимально простой и дешёвой в реализации, обеспечивала бы распознавание объектов в пространстве X с потерями, не превышающими N_0 .

2. Задано множество образов S , указана решающая функция D и допустимые затраты N_0 . Требуется найти такую систему признаков X , которая обладала бы достаточной информативностью при минимальных затратах на измерение этих признаков.

3. К третьему типу задач относятся задачи так называемой «таксономии»: задано множество реализаций образов в пространстве признаков X . С помощью решающей функции D нужно разделить на такое количество S , чтобы потери не превышали бы заданной величины N_0 .

Под *информативностью системы признаков* понимается совокупность различительных качеств или признаков, характеризующих определенный образ.

Часто задачу третьего типа обозначают термином «самообучение» или «обучение без учителя», а задачи первого и второго типа – «обучением с учителем» [31]. Поэтому задачи распознавания первого и второго типа состоят из процедур обучения и распознавания. Обучение с учителем заключается в показе изображений с эталонной информацией о расположении изображений объектов целевых классов на исходном изображении. В результате обучения распознающая система должна приобрести способность одинаково реагировать на различные изображения одинаковых образов и различно на изображения различных образов. Распознавание – это процесс уже обученной распознающей системы.

1.3 Критерий малоразмерности изображений

Существует большое разнообразие направлений, где востребовано обнаружение и распознавание объектов реального мира, расположенных на земной поверхности стационарными или передвижными системами, которые тоже расположены на земной поверхности. Одним из главных требований к таким системам является заблаговременное обнаружение и распознавание объектов. Для обеспечения заблаговременного распознавания объект интереса должен находиться на дальнем расстоянии от места фиксации. В оптических каналах объекты на дальних расстояниях представлены малоразмерными изображениями.

Малоразмерность изображений объектов можно определить с точки зрения вероятности распознать изображение объекта наблюдателем (оператором) через устройство наблюдения [32, 33] (рисунок 1.1).



Рисунок 1.1 – Функциональная схема процесса обнаружения объекта

Одной из первых работ, посвященных исследованию модели восприятия наблюдателя распознавать изображения объектов через устройство видения, была

выполнена Джоном Джонсоном в 1950-х годах. Джон Джонс изучал зависимость вероятности обнаружения и распознавания объектов от соответствующего размера изображения. Джонс получил эмпирические характеристики вероятностей обнаружения и распознавания изображений объектов от количества периодов H_p черно-белой штриховой миры, эквивалентной минимальному размеру изображения объекта или критическому размеру $H_{кр}$:

$$P = \frac{\left(\frac{H_p}{H_{p50}}\right)^E}{1 + \left(\frac{H_p}{H_{p50}}\right)^E}, \quad (1.3)$$

$$P = 2.7 + 0.7 \frac{H_p}{H_{p50}}, \quad (1.4)$$

где H_{p50} – количество периодов, при котором вероятность распознавания изображения объекта составляет 50 %. Для задачи обнаружения изображения объекта экспериментально получено $H_{обн.50} = 2$, а для задачи распознавания объекта $H_{расп.50} = 7$. Такой подход к определению вероятности является упрощённым, так как не учитывает факторы контрастности цели и неоднородности фона по яркости. Один период штриховой миры содержит два элемента миры со штрихами равной толщины – чёрной и белой (рисунок 1.2).

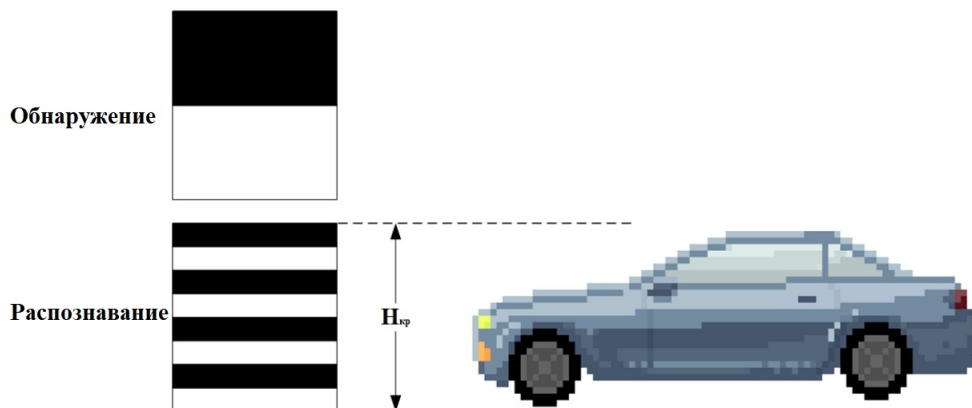


Рисунок 1.2 – Пример изображения штриховой миры, эквивалентной изображению объекта

Определим критерий малоразмерности изображения объекта. Минимальный размер изображения объекта находится в диапазоне 8 – 14 пикселей, что соответствует распознаванию с вероятностью 30 – 50 % по методу Джонсона.

Сам объект должен иметь большие физические размеры и находиться на дальних расстояниях. Размеры изображения объекта зависят от характеристик сенсорной аппаратуры (разрешение матрицы камеры, угол обзора видеокамеры, фокусное расстояние и др.) и внешних условия (физические размеры объекта, дальность до объекта и др.). Рассмотрим гипотетический пример фиксации изображения объекта класса «грузовой автомобиль» видеокамерой в условиях прямой видимости, и рассчитаем дальность до него (рисунок 1.3 а) на основе заданных параметров: разрешение матрицы по горизонтали M_{Γ} и $M_{\text{в}}$ составляет 960 пикселей, угол обзора видеокамеры по вертикали $\beta_{\text{в}}$ и горизонтали β_{Γ} составляет 4° , минимальный размер изображения (по вертикали) объекта $M_{\text{кр}}$ для вероятности распознавания 50 % составляет 14 пикселей, который соответствует физическому размеру объекта $G_{\text{об}} = 2.5$ м (высота объекта). С учётом указанных значений дальность до объекта $G_{\text{д},50}$ составляет:

$$G_{\text{д},50} \approx \frac{G_{\text{об}}}{2 \operatorname{tg} \frac{\varphi_{\text{об}}}{2}} \approx \frac{G_{\text{об}}}{2 \operatorname{tg} \frac{\beta_{\text{в}} M_{\text{об}}}{2 M_{\text{в}}}} \approx 2864 \text{ м}, \quad (1.5)$$

где $\varphi_{\text{об}}$ – угловой размер объекта; $M_{\text{об}}$ – размер объекта в пикселях.

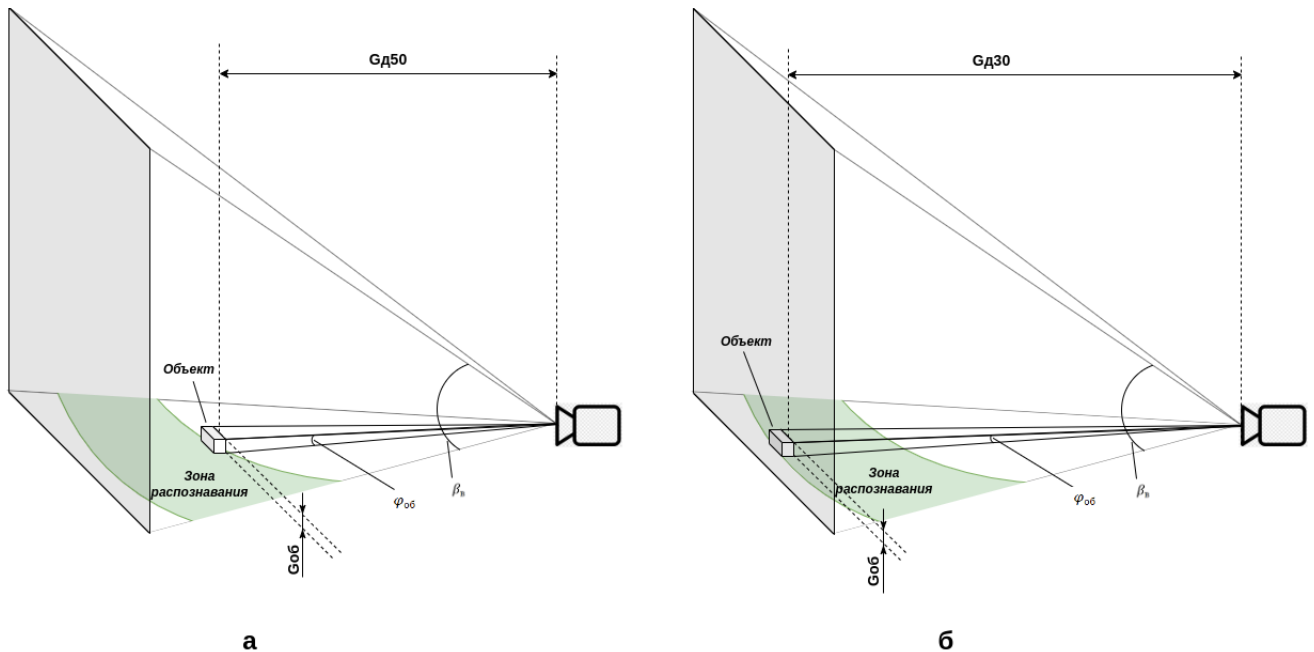


Рисунок 1.3 – Иллюстрация процесса фиксации изображений видеокамерой

Для вероятности распознавания 30 % минимальный размер изображения составляет 8 пикселей (рисунок 1.3 б). Дальность до объекта $G_{\text{д},30}$ с учётом фор-

мулы (1.5) составляет 4297 м. Для оценки качества распознавания изображений объектов используются специальные критерии, которые рассмотрены ниже.

1.4 Критерии качества в задаче распознавания

Ошибка распознавания алгоритма состоит из ошибки обнаружения и ошибки классификации. Для оценки ошибки обнаружения изображения объекта используется коэффициент Джаккара или Intersection Over Union (*IOU*) [34, 35], который характеризует меру степени пересечения между двумя обрамляющими прямоугольниками:

$$IOU = \frac{S_{\cap}}{S_{\cup}}, \quad (1.6)$$

где S_{\cap} – площадь пересечения между эталонным и предсказанным обрамляющими прямоугольниками, S_{\cup} – площадь объединения между эталонным и предсказанным обрамляющими прямоугольниками (рисунок 1.4).

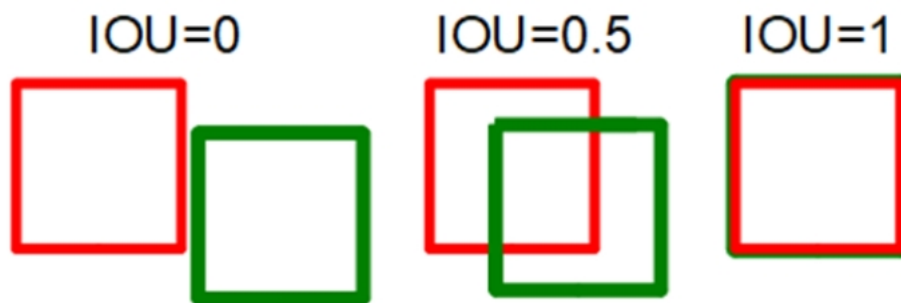


Рисунок 1.4 – Визуализация эталонного и предсказанного обрамляющих прямоугольников для разных значений коэффициента Intersection Over Union

Значения *IOU* варьируются в диапазоне от 0 до 1. Чем выше значение *IOU*, тем меньше ошибка обнаружения изображения объекта. Обычно пороговое значение *IOU* приравнивают к 0.5. Если $IOU < 0.5$, то считается как ложное обнаружение, в противном случае – как истинное обнаружение.

Критерии качества классификации в задаче распознавания рассчитываются исключительно на основе четырёх возможных результатов сопоставления экспертной оценки с решением алгоритма (предиктора) на тестовом наборе данных: *TruePositive* (*TP*) – количество истинно позитивных срабатываний, *FalsePositive* (*FP*) – количество ложно позитивных срабатываний, *FalseNegative* (*FN*) – количество ложно негативных срабатываний, *TrueNegative* (*TN*) – количество истинно негативных срабатываний (таблица 1).

Таблица 1 – Матрица несоответствий (ошибок) алгоритма бинарной классификации

Выход предиктора	Экспертная оценка	
	True	False
True	TP	FP (ошибка первого рода)
False	FN (ошибка второго рода)	TN

Данные оценки являются абсолютными ненормированными величинами и связаны с такими понятиями классической теории вероятностей, как вероятность ошибки первого и второго рода. Основными характеристиками качества распознающей системы являются показатели ошибок классификации – объём ложно позитивных и ложно негативных решений. Их производными являются вероятность правильного распознавания P_{pac} и частота ложной тревоги $F_{\text{лт}}$ [36, 37]. В условиях ограниченного числа опытов частота ложной тревоги оценивается как доля ложно позитивных решений FP во всей выборке:

$$F_{\text{лт}}|_{FP \leq K_{\text{кадр}}} = \frac{FP}{K_{\text{кадр}}}, \quad (1.7)$$

где $K_{\text{кадр}}$ – количество изображений в выборке. Зная период выдачи кадров камерой, можно оценить частоту появления ложной тревоги во времени. Для фиксированной частоты ложной тревоги оценивается вероятность правильного распознавания $P_{\text{pac}}|_{F_{\text{лт}}=\text{const}}$ как долю истинно позитивных решений среди всех позитивных образцов:

$$P_{\text{pac}}|_{F_{\text{лт}}=\text{const}} = \frac{TP}{TP + FN}, \quad (1.8)$$

Классическими критериями качества классификации являются *Accuracy*, который характеризует долю правильных решений алгоритма среди всех решений:

$$Accuracy = \frac{TP}{TP + FP + FN} \quad (1.9)$$

и критерий *ErrorRate* (частота ошибок), который характеризует долю ошибок алгоритма среди всех решений:

$$ErrorRate = \frac{FP + FN}{TP + FP + FN}. \quad (1.10)$$

Еще одними производными показателей ошибок классификации являются количественные характеристики качества распознавания Полнота (*Recall*) и Точность (*Precision*). *Precision* характеризует долю правильно предсказанных позитивных решений среди всех позитивных решений алгоритма для одного класса:

$$Precision = \frac{TP}{TP + FP}. \quad (1.11)$$

Recall отражает долю истинно позитивных решений среди всех образцов одного класса:

$$Recall = \frac{TP}{TP + FN}. \quad (1.12)$$

Recall учитывает ошибку распознавания второго рода, а *Precision* ошибку распознавания первого рода. По своей сути *Recall* является оценкой вероятности правильного распознавания.

Варьированием порога уверенности решения алгоритма формируется зависимость *Precision – Recall*. На основе зависимости функции *Precision – Recall* можно вычислить критерий Average Precision (*AP*), который характеризует качество распознавания одного класса *AP* и равен площади под кривой *Precision – Recall* (рисунок 1.5) [38, 39].

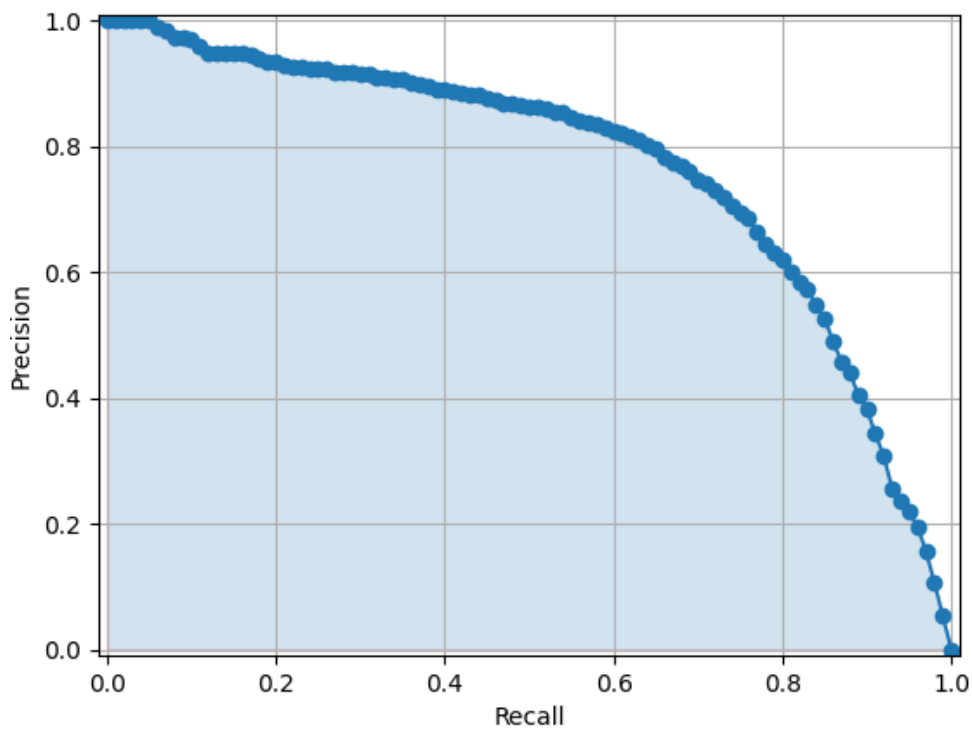


Рисунок 1.5 – Пример типовой зависимости *Precision – Recall*

Критерием качества распознавания всех классов является интегральная характеристика *MeanAveragePrecision (mAP)*, которая рассчитывается как:

$$mAP = \frac{\sum_{i=1}^k AP_i}{k}, \quad (1.13)$$

где k – количество классов.

1.5 Методы и алгоритмы решения задачи распознавания изображений

Классические подходы, решающие задачу распознавания изображений объектов, состоят из двух этапов: формирование различных областей изображения, например, с помощью скользящих окон разных размеров и последующей классификации каждой области. Классификация классическими подходами основана на построение усиленных решающих функций в виде линейных комбинаций слабых решающих функций. Большинство алгоритмов строят решающую функцию на основе набора признаков, выбранного разработчиком. Набор признаков определяется характером объекта, который нужно распознать. Если изображения

объектов содержат кусочно-линейные геометрические элементы, они могут быть представлены с помощью контуров и границ, выделение которых опирается на дескрипторы особых точек, углов, границ и др. Выделение признаков объектов, содержащих более сложные структуры, можно использовать разложение области изображения по системе базисных функций или свёртка с фиксированным банком фильтров.

Главным фактором, влияющим на качество распознавания, является различительная способность набора признаков классов. Большая различительная способность набора признаков обеспечивает построение решающей функции классификатора с большей различительной способностью. И наоборот, чем меньше различительная способность набора признаков, тем хуже разделяющая способность решающей функции. Поэтому выбор набора признаков является нетривиальной задачей. Эту проблему можно отнести к основному недостатку классических методов.

Достоинством данных подходов является возможность строить компактные решающие функции, обеспечивая высокое быстродействие алгоритма на устройствах с ограниченным вычислительным ресурсом.

С увеличением вычислительных мощностей и появлением больших объёмов обучающих данных в открытом доступе стали появляться алгоритмы глубокого обучения на основе свёрточных нейронных сетей (СНС). В 2012 году вырос интерес к СНС из-за появления архитектуры СНС AlexNet [40], которая показала меньшую ошибку классификации на крупномасштабном датасете ImageNet [41, 42]. Успех применения СНС определил вектор развития алгоритмов распознавания изображений вплоть до настоящего времени.

Входными данными для СНС являются изображения, n -мерные тензоры изображений (батч), пирамиды изображений. Состав современных алгоритмов распознавания изображений на основе СНС состоит из экстрактора признаков (Backbone) и декодера признаков (Head) [7] (рисунок 1.6). Для повышения сходимости СНС, обучение алгоритмов начинают с инициализации весовых коэффициентов предварительно обученным экстрактором признаков [43, 44].

Обучение экстрактора признаков проводится на крупномасштабных классификационных датасетах, таких как ImageNet и др.

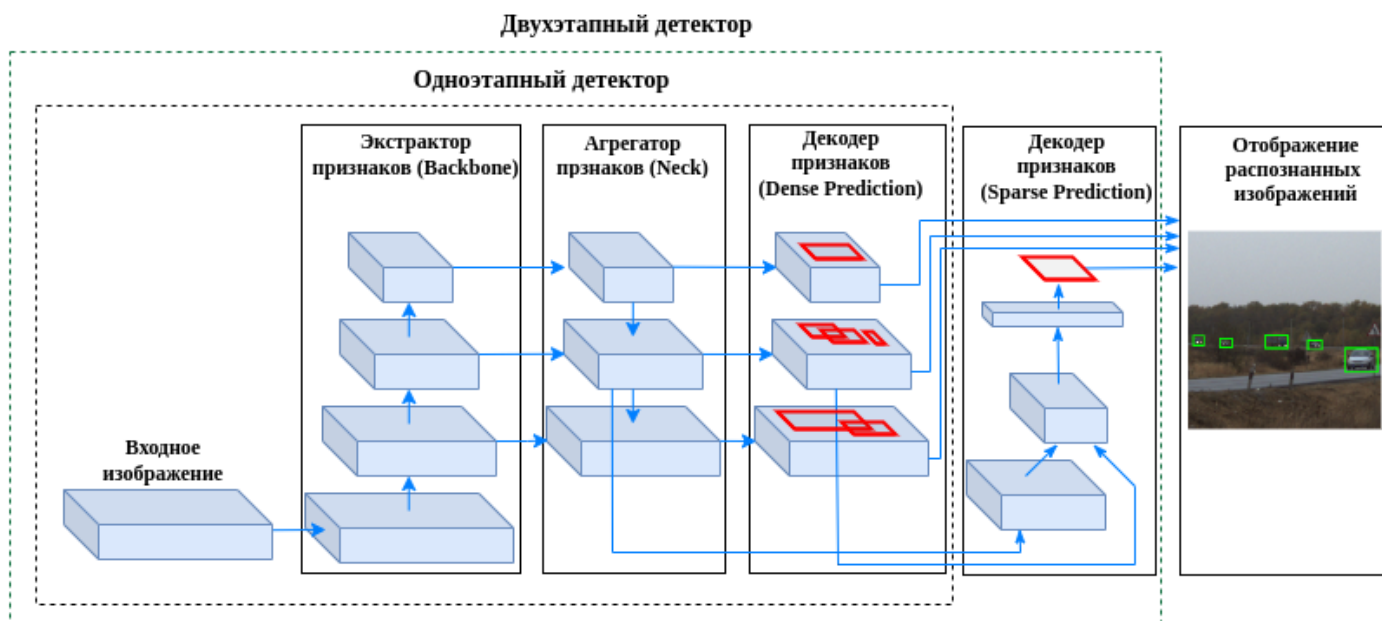


Рисунок 1.6 – Структурная схема алгоритмов распознавания изображений объектов на базе свёрточных нейронных сетей

Обработка изображений вычислительно сложными архитектурами СНС в масштабе реального времени возможна только на устройствах с нейронным ускорителем типа графический процессор Graphics Processing Unit (GPU), тензорный процессор Tensor Processing Unit (TPU) и др. Основопологающей операцией в СНС является «двухмерная свёртка» (Convolution 2D), на базе которой основаны современные архитектуры СНС: VGG [45], ResNet [46], ResNeXt [47] или DenseNet [48] и др.

Вычислительные устройства без нейронных ускорителей обрабатывают данные на центральном процессоре Central Processing Unit (CPU), что ограничивает применение архитектур СНС, содержащих сложные операции. В таких ситуациях выбирают архитектуры СНС, в основе которых лежит операция «двухмерная поканальная свёртка» (Depth wise Convolution 2D): SqueezeNet [49], MobileNet [50–53] или ShuffleNet [54, 55].

Современные архитектуры СНС часто используют дополнительные слои между экстрактором признаков и декодером признаков для того, чтобы собрать семантическую информацию с разных масштабов пирамиды признаков. Такое

объединение принято называть агрегатором признаков (Neck). К распространённым методам агрегирования признаков можно отнести: Feature Pyramid Network (FPN) [56], Path Aggregation Network (PAN) [57], BiFPN [58] и NAS-FPN [59].

Методы декодирования агрегированных признаков условно можно разделить на одноэтапные (Dense Prediction) и двухэтапные (Sparse Prediction) подходы. Двухэтапные подходы решают задачу распознавания в два этапа. На первом этапе формируются регионы кандидатов, а на втором этапе происходит классификация сформированных регионов. Одноэтапные подходы проводят распознавание, не используя предложения регионов, а напрямую распознают объекты. Двухэтапные подходы обеспечивают более высокое качество распознавания, но вычислительно сложнее в отличие от одноэтапных подходов. Это отличие делает одноэтапные подходы более «привлекательными» для применения в устройствах с ограниченным вычислительным ресурсом.

Одноэтапные и двухэтапные подходы основываются на априорной информации о размерах изображений объектов. Привязка алгоритмов к априорной информации является недостатком, потому что такой подход приводит к неудовлетворительному качеству распознавания модели СНС на новых изображениях объектов, имеющих отличные пиксельные размеры.

Из числа двухэтапных алгоритмов, использующих априорные данные, можно выделить fast R-CNN [60], R-CNN [61], faster R-CNN [62], R-FCN [63] и Libra R-CNN [64]. Также есть двухэтапные алгоритмы без привязки к априорной информации, например, RepPoints RepPoints [65].

К числу одноэтапных подходов, использующих априорную информацию о размерах изображений, относятся You Only Look Once (YOLO) [14, 66, 67], Single Shot Multibox Detector (SSD) [15] и RetinaNet [68]. У одноэтапных СНС есть реализации алгоритмов без привязки к априорным данным: CenterNet [16, 69], CornerNet [70, 71], FCOS [18], DETR [72] и др.

Обобщая все сказанное: выполнение СНС опирается на следующие компоненты, приведенные в таблице 2.

Таблица 2 – Составные компоненты распознавания изображений свёрточными нейронными сетями

Составная часть СНС	Реализация составной части СНС
Входные данные	Изображения, батч, пирамиды изображений.
Извлечение признаков (Backbone)	VGG, ResNet-50, SpineNet [73], ResNeXt, DenseNet, SqueezeNet, MobileNet, ShuffleNet, CSPResNeXt50 [74], CSPDarknet53 [74] и др.
Агрегирование признаков (Neck)	<ol style="list-style-type: none"> 1. Additional blocks: SPP [75], ASPP [76], RFB [77], SAM [78] и др.; 2. Path-aggregation blocks: FPN, PAN [57], NAS FPN [59], BiFPN [58], ASFF [79], SFAM [80] др.
Декодирование признаков (Head)	<ol style="list-style-type: none"> 1. Dense Prediction (одноэтапные подходы): RPN, SSD, YOLO, RetinaNet, CornerNet, CenterNet, MatrixNet, FCOS др.; 2. Sparse Prediction (двухэтапные подходы): Faster R-CNN, R-FCN, Mask R-CNN, RepPoints др.

Проблемой СНС является возможность предсказать несколько ложных решений для одного и того же объекта на изображении. Это приводит к повышению количества ошибок распознавания. Решение этой проблемы основано на применении алгоритмов постобработки выходных данных. Один из зарекомендованных алгоритмов постобработки выходных данных является алгоритм Non Maximum Suppression (NMS). Исходный метод NMS не учитывает уверенность решения алгоритма, поэтому автор [61] добавил оценку достоверности решения для RCNN. Таким образом, происходит фильтрация обрамляющих прямоугольников по IOU в порядке убывания уверенности решения по IOU.

Достоинством СНС является то, что в процессе обучения заранее определенной архитектуры СНС происходит подбор коэффициентов фильтров таким

образом, чтобы минимизировать функцию потерь, заданную разработчиком. К недостаткам СНС можно отнести:

1. Нет чётких правил построения архитектур СНС.
2. Процесс разработки архитектуры СНС является эвристическим процессом, да и результаты обучения не гарантируют нахождения оптимального и единственного решения.
3. Большое количество настраиваемых параметров замедляет обучение и повышает вероятности попадания в локальные минимумы в процессе градиентного спуска.
4. Обучение СНС требует больших объёмов тренировочных данных.
5. Обучение СНС требует специализированных вычислительных серверов с массивами графических процессоров.

Большинство известных архитектур СНС разрабатывались с целью минимизации ошибок распознавания на крупномасштабных датасетах. Для решения частных задач с более узким набором классов вычислительная сложность архитектур может оказаться избыточной.

1.6 Переобучение свёрточных нейронных сетей в задаче распознавания малоразмерных изображений

За последние несколько лет алгоритмы обработки изображений на основе СНС достигли значительного прогресса на открытых датасетах, однако в решении практических задач не всегда получается достичь требуемых характеристик распознавания. Вычислительно сложные модели многослойных нейронных сетей (НС), такие как СНС, способны распознавать едва различимые изображения объектов, но распознавание малоразмерных изображений вызывает трудности даже у передовых архитектур СНС. Это связано с тем, что СНС обучаются извлекать шумовые признаки. *Модель* СНС есть программная реализация архитектуры СНС для электронных вычислительных машин.

В процессе обучения СНС автоматически формирует гиперплоскость, разделяющую образы в пространстве признаков с помощью определения весовых коэффициентов. Обучение СНС на невалидных данных ведёт к формированию решающей функции классификатора, которая не способна определить некую скрытую взаимосвязь между входными и выходными данными. Такую взаимосвязь называют *обобщающей способностью* [81]. В большинстве случаев проблема обобщения данных моделью НС связана с переобучением. *Переобучение* – это негативный эффект, возникающий, когда алгоритм в процессе обучения формирует решающую функцию, слишком близко соответствующую тренировочному набору данных. Переобученная модель НС обеспечивает высокие показатели качества распознавания на данных, которые входили в тренировочную выборку, а на тестовых данных, которые модель НС никогда «не видела», показывает отрицательный результат. Это связано с тем, что объём различительных признаков, характеризующих малоразмерные изображения объектов, недостаточен для успешного разделения образов.

На рисунке 1.7 приведена возможная зависимость значений ошибки обучения НС на тренировочном и тестовом выборках от эпох. Красная кривая иллюстрирует значение ошибки обучения НС на тренировочной выборке, а зеленая кривая – значение ошибки обучения НС на тестовой выборке. На начальном этапе обучения значения ошибок обучения на тренировочной и тестовой выборках уменьшается. На этом этапе модель НС находится в состоянии недообучения. После определенного момента ошибка обучения на тестовой выборке начинает повышаться, а на тренировочной выборке продолжает уменьшаться. После этой эпохи обучения модель НС начинает переобучаться. В момент обучения, когда ошибка на тестовой выборке начнет увеличиваться, модель НС является квазиоптимальной и обладает лучшей разделяющей способностью.

Рассмотрим пример выполнения условий решающей функции НС в задаче классификации. Для гипотетического примера на рисунке 1.8 представлено двухмерное евклидово пространство, содержащее векторы линейно разделимых объектов (красные и зеленые точки). Вектор весов w_i представляется набором

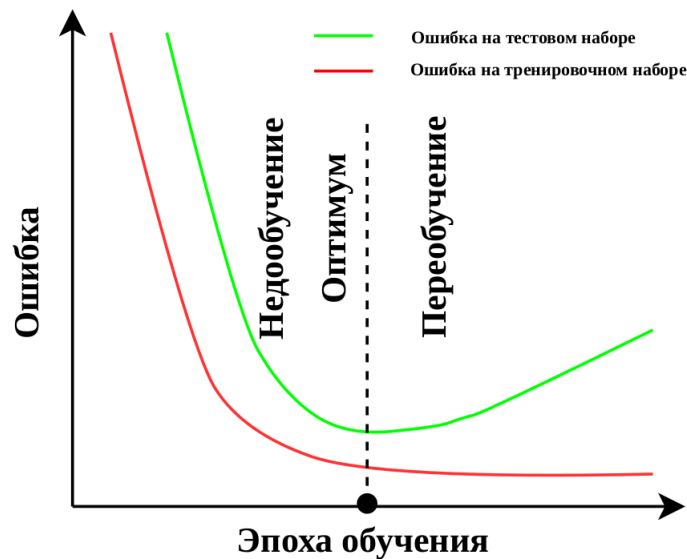


Рисунок 1.7 – Иллюстрация процесса обучения модели нейронной сети

весовых коэффициентов, который формируется в процессе обучения НС на заданной тренировочной выборке и определяет разделяющую кривую (синяя кривая). Пусть $D(X) = w_1x_1 + w_2x_2 + w_3 = 0$ – уравнение разделяющей прямой НС, где x_i – входные данные (признаки). Из рисунка видно, что подстановка признаков объекта класса B_1 даст отрицательные значения (выполняется условие $D(X) < 0$), а класса B_2 – положительные значения (выполняется условие $D(X) > 0$). Таким образом, функцию $D(X)$ можно использовать в качестве решающей функции для классификации объектов двух и более классов.

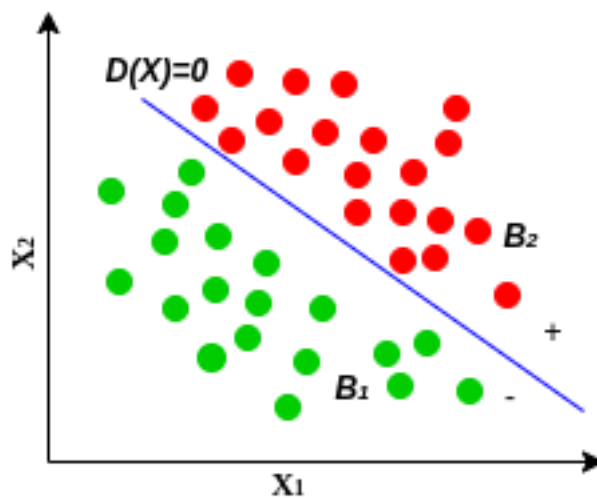


Рисунок 1.8 – Отображение множества объектов двух классов в виде точек в двухмерном признаковом пространстве

Если границы образов в пространстве признаков пересекаются, то образуются зоны неоднозначности, которые обусловлены неполным признаковым описанием объектов, наличием шумовых признаков, ошибками в разметке и др. Обучение НС на данных, представленных с зонами неоднозначности, может постепенно приводить к переобучению модели. На рисунках 1.9 представлены разделяющие поверхности НС на разных этапах обучения, объекты тренировочных и тестовых данных в двухмерном признаковом пространстве. Переобученная НС безошибочно классифицирует объекты тренировочных данных. Однако классификация объектов тестовых данных приводит к противоположному результату, потому что модель адаптировала разделяющую плоскость только для объектов тренировочных данных (рисунок 1.9 в, д). А недообученная модель ещё не сформировала весовые коэффициенты в процессе обучения для успешной классификации объектов (рисунок 1.9 а, г). Можно заметить, что меньшую ошибку классификации на тестовых данных обеспечивает модель, разделяющая плоскость, которая представлена на рисунках (рисунок 1.9 б, д). В данном контексте модель сформировала компоненты вектора весов, которые определили оптимальную разделяющую границу для классификации объектов.

Переобучение СНС в задаче распознавания малоразмерных изображений обусловлено малым количеством различительных признаков, что приводит к появлению зон неоднозначности, как следствие, приводя к переобучению. Кроме того, изображения малоразмерных объектов в большей степени подвержены деформациям из-за перекрытия другими объектами, что уменьшает количество различительных признаков. Современные архитектуры СНС содержат миллионы обучаемых параметров, и с каждым годом вычислительные сложности СНС продолжают расти в угоду минимизации ошибки распознавания, что может ускорить переобучение СНС в задаче распознавания малоразмерных изображений. Предотвращение переобучения моделей СНС является одной из главных задач машинного обучения, которая объединила множество методов, названных термином *регуляризация*.

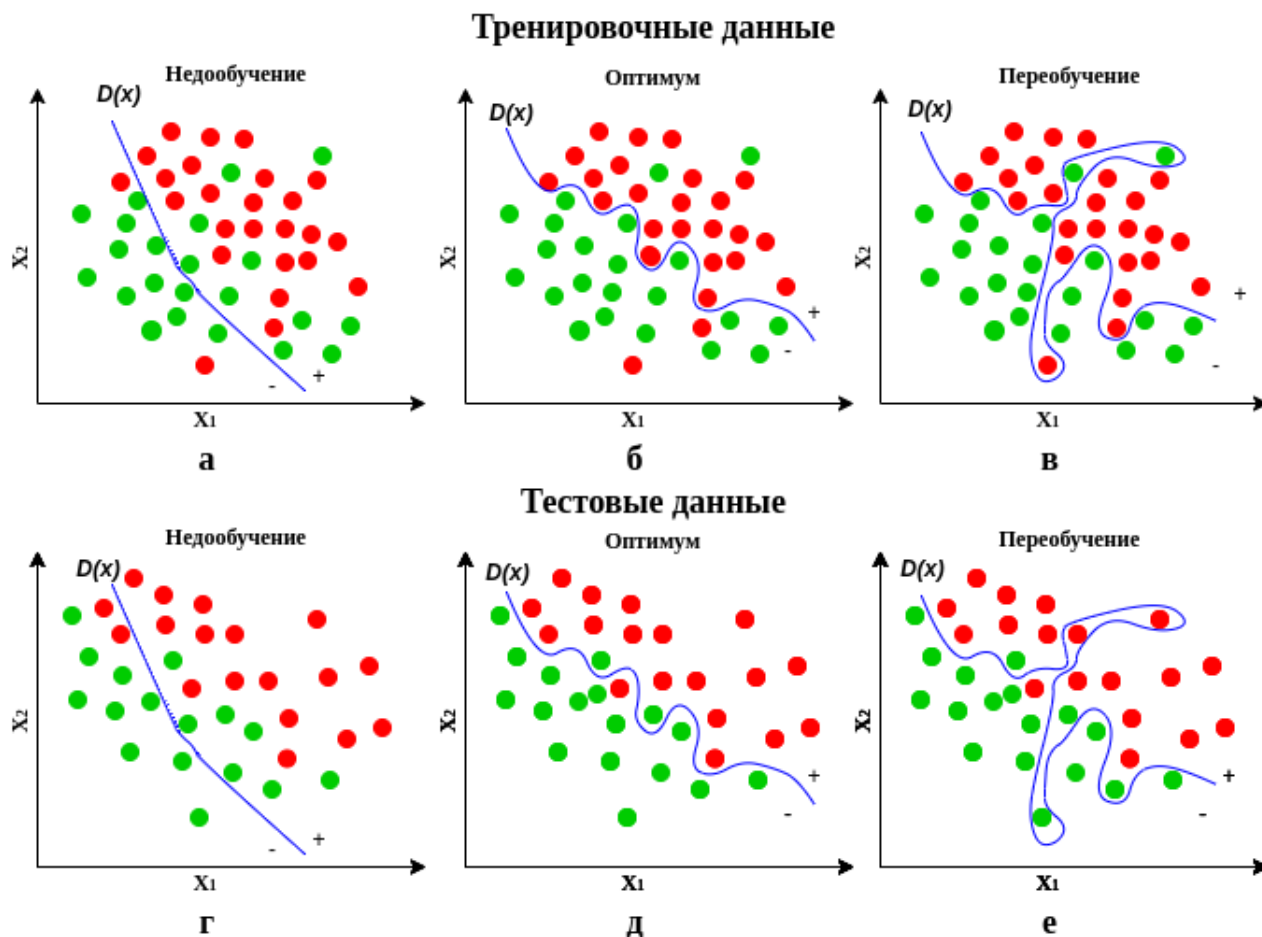


Рисунок 1.9 – Иллюстрация решающей функции нейронной сети на разных этапах обучения

1.7 Регуляризация свёрточных нейронных сетей

В широком смысле регуляризация обобщает широкий спектр подходов, которые предотвращают переобучение НС. С точки зрения машинного обучения весь спектр подходов регуляризации можно разделить на три обширные группы:

1. Изменение функции потерь.
2. Изменение процесса обучения и архитектуры нейронной сети.
3. Изменение данных.

Далее рассмотрим каждую группу подходов регуляризации моделей НС подробнее.

Изменение функции потерь. Переобученная модель НС имеет решающую функцию, которая относит объекты к определенному классу с минимальной ошибкой на тренировочном наборе данных. Такая решающая функция не будет

обобщать новые данные. В процессе переобучения модель НС подстраивается под тренировочные данные, что приводит к формированию весовых коэффициентов с большими значениями. Таким образом, модель НС пытается минимизировать ошибку на тренировочном наборе данных. В таких случаях применяют регуляризацию функции потерь, которая ограничивает величину весовых значений модели, добавляя штраф для всех весов к функции ошибки модели. Существует несколько видов регуляризации, ниже приведены два самых популярных вида регуляризации функции потерь.

Регуляризация L_1 или Lasso (Least Absolute Shrinkage and Selection Operator) – регуляризация, добавляет штраф, равный абсолютному значению коэффициентов модели:

$$L_1 = L + \lambda \sum_i |w_i|, \quad (1.14)$$

где L – ошибка модели по выбранной целевой функции; λ – коэффициент регуляризации; w – значение весового коэффициента. Регуляризация L_1 склонна к селекции признаков, так как она может обнулить весовые коэффициенты.

Регуляризация L_2 или Ridge Regression, так же как и регуляризация L_1 , добавляет штраф к целевой функции:

$$L_2 = L + \lambda \sum_i (w_i)^2. \quad (1.15)$$

Регуляризация L_2 штрафует большие значения весовых коэффициентов, тем самым снижая их значимость. В отличие от регуляризации L_1 , весовые коэффициенты не полностью приравниваются к нулевым значениям, а лишь приближают их к нулю.

Основное преимущество применения регуляризации функции потерь в том, что она часто приводит к формированию моделей с более высоким качеством распознавания, главным же её недостатком является введение дополнительного гиперпараметра. Гиперпараметры – это параметры, которые задаются до обучения модели НС и не изменяются в процессе обучения. *Гиперпараметры* могут

являться параметрами, которые управляют топологией модели НС или параметрами алгоритма обучения. К таким параметрам можно отнести количество скрытых слоев, количество фильтров в операции «свёртка», тип оптимизатора и др.

Изменение процесса обучения и архитектуры нейронной сети. Внесение определенных изменений в архитектуру НС или процесс обучения является распространенным способом регуляризации, который позволяет сделать НС более устойчивой к переобучению.

Ранняя остановка обучения является одним из подходов изменения процесса обучения НС. Применение ранней остановки заключается в остановке обучения модели НС до того, как она начнет переобучаться. Так как НС обучаются итеративно методом обратного распространения ошибки, то для каждой эпохи обучения можно сохранить и выбрать модель НС, удовлетворяющую необходимым критериям качества распознавания. Современные фреймворки машинного обучения, такие как *PyTorch*, *TensorFlow*, *Keras* и др., предоставляют такую возможность.

Вычислительная сложность современных нейросетевых алгоритмов с каждым годом увеличивается, что позволяет НС обобщить большие объёмы данных. Но с другой стороны, такая гибкость модели НС предрасположена к переобучению на небольших данных, в том числе, где объекты представлены малым количеством признаков. Поэтому в определённых случаях для предотвращения переобучения модели НС целесообразно уменьшить вычислительную сложность архитектуры сети. Как правило, это достигается за счет сокращения скрытых слоёв или количества нейронов. Так же может быть выбрана другая архитектура НС с меньшей вычислительной сложностью.

Одним из популярных методов регуляризации НС является Dropout [82, 83]. Алгоритм заключается в следующем: на каждом шаге обучения каждый нейрон с определенной вероятностью может быть временно исключен. Таким образом, этот нейрон полностью игнорируется на этом этапе обучения, но на следующем шаге обучения этот нейрон будет активным. Как правило, вероятность отключения нейрона равна 50 %. После обучения, в режиме распознавания, нейроны

не отключаются. Применение Dropout позволяет получить модель НС с большей обобщаемой способностью.

Обучение НС усложняется тем, что распределение входных данных каждого слоя меняется в процессе обучения. Авторы статьи [84] называют это явление внутренним ковариационным сдвигом и предлагают решение этой проблемы, нормализуя входной батч для каждого слоя НС. Применение такого подхода назвали *батч-нормализацией* (Batch-normalization). Кроме того, что нормализация позволяет сократить время обучение, так она ещё и выступает регуляризатором. Результат работы НС зависит не от одного экземпляра, а от всего батча, не позволяя адаптировать решение, что увеличивает обобщение НС.

Нормализация является частью архитектуры НС и выполняется в процессе распознавания. При обучении НС с батч-нормализацией обрабатываемый образец рассматривается в сочетании с другими образцами в пакете, и модель больше не выдает детерминированные значения для данного тренировочного примера. Этот эффект позволяет повысить способность к обобщению модели НС.

Изменение данных. Зачастую проблема переобучения связана с данными, которые не содержат достаточного количества признакового описания объектов для того, чтобы модель НС приобрела способность к обобщению [85]. Чтобы улучшить обобщающую способность модели НС, нужно иметь большой объём разнообразных, репрезентативных данных. Расширение тренировочного набора данных за счет преобразования исходного набора данных искусственным образом является популярным методом регуляризации, который получил название *аугментация данных*. Искусственное увеличение набора данных изображений достигается за счет различных преобразований над исходными изображениями так, чтобы изображения объектов не выходили за рамки визуального представления образа. Можно выделить одни из часто используемых преобразований изображений [7]:

1. Трансформация изображений: поворот, смещение по ширине и высоте, сдвиг изображения, масштабирование, горизонтальное или вертикальное зеркальное отражение, вырезание части изображения и др.

2. Яркостные преобразования: изменение яркости всего изображения или определенных каналов изображения в различных цветовых пространствах, гамма-коррекция, размытие, добавление шума и др.

3. Мозаичное преобразование (mosaic) – это новый метод увеличения данных, который формирует новое составное изображение на основе четырёх отдельных изображений из набора тренировочных данных и др.

Еще одним методом преобразования данных является фильтрация шума [86]. В процессе приёма и передачи данных появляются различные шумы, связанные с ошибкой квантования, шумами $p - n$ -перехода в полупроводниковых элементах, наводками в каналах связи и др. Для устранения шума в данных применяются как стандартные методы фильтрации, так и специализированные методы, основанные на нелинейной фильтрации [87–90]. Применение фильтрации шума может носить как локальный, так и глобальный характер.

Расширение признакового пространства. В последнее время большую популярность набирают методы регуляризации моделей СНС, основанные на расширении признакового пространства с помощью комплексирования разномодальных данных. Разномодальные данные состоят из нескольких изображений различных типов: изображения видимого оптического диапазона, ИК-изображения, изображения лазерных дальномеров, радиолокационные изображения, видеокадры с признаками движения и др. Под комплексированием понимается слияние двух и более изображений различных типов в одно с помощью специальных методов. Можно выделить основные методы комплексирования: преобразование Фурье, пирамидальное преобразование, вейвлет-преобразование, слияние на уровне пикселей [91–94].

В научной работе [19] авторы предотвращают переобучение путём расширения исходного признакового пространства за счет комплексирования разномодальных данных. Разномодальные данные представлены в виде изображений различных спектральных диапазонов, которые применяются для решения практических задач.

В статьях [20, 21] авторы предотвращают переобучение СНС с помощью расширения исходного признакового пространства видимого оптического диапазона данными с лазерных дальномеров, ИК-изображениями и изображениями с признаком движения [22, 23]. Недостаток применения изображений с лазерных дальномеров, ИК-изображений, радиолокационных изображений заключается в установке на бортовых системах необходимых сенсоров, что усложняет конструкцию и увеличивает стоимость аппаратуры, в отличие от видеокadres с признаками движения, которые можно получить менее затратным способом: с помощью обработки последовательности изображений видимого оптического диапазона алгоритмами цифровой обработки изображений. Существующие алгоритмы формируют признаки движения объектов на основе анализа признаков, полученных в два момента времени. Таким образом, данные подходы не учитывают признаки объектов в процессе их изменения.

Проблема 1. Обучение свёрточных нейронных сетей на данных, представленных неинформативной системой признаков, приводит к ошибкам распознавания малоразмерных изображений, что требует эффективных методов, позволяющих увеличить количество различительных признаков.

Исходя из первой проблемы, вытекает вторая проблема, связанная с разработкой алгоритма, реализующего метод регуляризации СНС путем расширения исходного признакового пространства признаками движения, сформированными в процессе продолжительного наблюдения за объектами. Алгоритмы предварительной обработки изображений с целью комплексирования признаков движения не учитывают признаки объектов, сформированные в процессе продолжительного наблюдения.

Проблема 2. Современные системы обработки видеоданных предъявляют высокие требования к качеству распознавания малоразмерных изображений объектов, что требует новых алгоритмов предварительной обработки изображений, которые будут учитывать признаки объектов в процессе изменения.

Критерием информативности системы признаков является величина ошибок распознавания. Чтобы оценить информативность признаков, нужно иметь большие объёмы репрезентативных датасетов.

1.8 Датасеты для задачи распознавания изображений

Для сравнения алгоритмов распознавания изображений объектов традиционно используются контрольные тесты из открытых датасетов, содержащих тысячи размеченных вручную изображений. Для каждого изображения формируются эталонные значения, в которых содержится информация о расположении изображений объектов с меткой класса. В настоящее время для тестирования алгоритмов автоматического распознавания изображений объектов используются общепринятые контрольные тесты Pattern Analysis, Statistical Modeling and Computational Learning Visual Object Classes (PASCAL VOC) [38], ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [41], MS COCO.

Для каждого из перечисленных наборов данных ежегодно проводятся открытые международные соревнования, которые позволяют независимо оценить качество распознавания самых современных алгоритмов. В соревнованиях принимают участие сотрудники лучших компаний и студенты лучших университетов мира. Организаторы соревнований ежегодно формируют выборку из двух частей: открытую, которая используется участниками для разработки алгоритмов (в том числе и для процесса обучения), и закрытую часть, доступную только организаторам и используемую для оценки качества алгоритмов.

В настоящее время существуют датасеты аннотированных изображений различных типов для задачи распознавания изображений объектов:

1. Датасеты тепловых изображений. Такие датасеты состоят из изображений, полученных сенсорными датчиками в ИК-диапазоне. К числу таких датасетов можно отнести Thermal Imaging Dataset for Person Detection in Intrusion Warning Systems [95].

2. Разноспектральные датасеты изображений. Как правило, разноспектральные датасеты включают изображения, полученные в двух и более оптических диапазонов. Например, можно выделить разноспектральный датасет Visible-infrared Paired Dataset for Low-light Vision [96], который состоит из видимого и ИК оптических диапазонов.

3. Датасеты последовательностей видеокадров. Датасеты такого типа содержат видеокадры, на которых представлены изображения подвижных объектов. Каждый видеокадр имеет свой файл аннотации с информацией о локализации объектов в видеокадре и соответствующими метками классов. Можно выделить крупномасштабный датасет распознавания изображений объектов в последовательности видеокадров ImageNet VID.

Всё разнообразие датасетов для задачи распознавания не может удовлетворить различные узконаправленные задачи, например, задачу распознавания малоразмерных изображений объектов. Решить проблему дефицита данных можно с помощью мобильного стенда записи реальных видеоданных. Такой подход имеет недостатки, которые повышают стоимость разработки распознающей системы:

1. Специфика решаемой задачи может накладывать условия на географию местности, откуда будет проводиться съёмка видеосюжетов.
2. Дорогостоящая специализированная аппаратура.
3. Формирование и разметка данных требует специалистов узкого профиля.
4. Формирование и разметка данных – трудоёмкий процесс, который требует ресурса времени и др.

Обучение СНС на изображениях с расширенным признаковым пространством подразумевает применение разномодальных данных, использование которых приводит к повышению стоимости описания эталонов (формирование датасетов) $N(S)$. Затраты процедуры формирования датасетов включают в себя: стоимость записи разномодальных изображений, стоимость ручной разметки изображений, стоимость записывающего оборудования, а также стоимость ошибок в аннотациях, свойственных ручной разметки и др.

Проблема 3. Разработка методов и алгоритмов, решающих задачу распознавания малоразмерных изображений, нуждается в подходах, позволяющих снизить стоимость процедуры обучения распознающего алгоритма за счёт минимизации стоимости формирования датасетов разномодальных изображений.

В большей степени алгоритмы распознавания малоразмерных изображений находят своё применение при создании бортовых систем раннего обнаружения объектов, которые устанавливаются на автономные носители. Это накладывает ограничение на предоставляемую потребляемую мощность встраиваемых вычислительных устройств, где выполняются алгоритмы цифровой обработки изображений. Таким образом, сужается спектр устройств, возможных для применения.

1.9 Построение свёрточных нейронных сетей на устройствах с ограниченным вычислительным ресурсом

Основной объём требований в системах раннего распознавания изображений объектов предъявляется к качеству распознавания и времени выполнения алгоритмов цифровой обработки изображений, которые строятся на устройствах с ограниченным вычислительным ресурсом. На рисунке 1.10 представлена структурная схема системы распознавания изображений объектов. Под устройством с ограниченным вычислительным ресурсом понимается встраиваемое вычислительное устройство с малым форм-фактором, низким энергопотреблением, ограниченной производительностью, наличием аппаратно-программной поддержки центрального процессора и нейронного ускорителя (в рамках решения задач диссертационной работы). Под нейронным ускорителем понимается аппаратный вычислитель с программной поддержкой, на котором строятся и выполняются нейросетевые алгоритмы с применением подходов оптимизации вычислений.

Достижение требуемых вероятностных характеристик распознавания подразумевает применение вычислительно сложных алгоритмических решений на



Рисунок 1.10 – Структурная схема распознающей системы

базе СНС. Обработка данных СНС на устройствах с ограниченным вычислительным ресурсом должна осуществляться в масштабе реального времени. Для обеспечения требуемых временных характеристик выполнение вычислительно сложных этапов СНС осуществляется на вычислительных мощностях нейронного ускорителя (рисунок 1.11). Нейронные ускорители позволяют обрабатывать большое количество однотипных операций параллельно, что позволяет повысить скорость обработки данных.

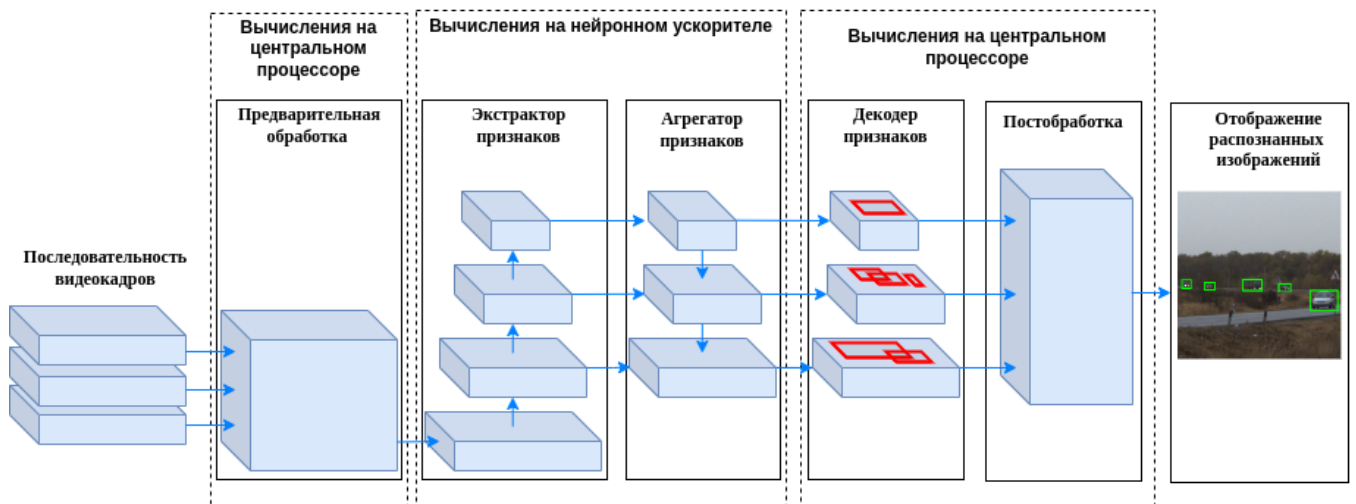


Рисунок 1.11 – Структурная схема распознавания изображений одноэтапными свёрточными нейронными сетями на устройствах с ограниченным вычислительным ресурсом

В настоящее время большой популярностью пользуются вычислительные устройства с аппаратной поддержкой нейронного ускорителя на базе графического и нейронного процессоров, которые рассмотрены ниже.

Снижение ошибки распознавания малоразмерных изображений объектов с помощью использования новой более информативной системы признаков (решение задачи поиска системы информативных признаков) может осуществляться алгоритмами предварительной обработки данных. Алгоритмы предварительной обработки данных формируют новое изображение, в котором учтено больше различительных признаков за счет комплексирования разномодальных изображений. Применение алгоритмов предварительной обработки изображений повышает стоимость процедуры измерения признаков $N(X)$ за счёт увеличения затрат на обработку данных. А предварительная обработка изображений высокого разрешения на центральном процессоре может приводить к неприемлемой временной задержке выполнения алгоритма в целом.

Проблема 4. Выполнение алгоритмов предварительной обработки изображений на центральном процессоре устройств с ограниченным вычислительным ресурсом требует эффективных алгоритмических и программных решений, позволяющих сократить временную задержку выполнения алгоритма.

1.10 Выбор вычислительного устройства для построения свёрточных нейронных сетей на устройствах с ограниченным вычислительным ресурсом

Обычно алгоритмы компьютерного зрения на базе СНС применяются в интеллектуальных системах, где требуется локальная встроенная цифровая обработка данных. Гаскаров Д. В. [97, 98] определяет интеллектуальную систему как аппаратно-программный комплекс с интеллектуальной поддержкой при решении задач с участием оператора и без оператора. Другими словами, способность системы имитировать когнитивные функции человека [99].

СНС выполняют функцию распознавания интеллектуальной системы. Распознающие системы располагаются на автономных носителях, поэтому вычислительные устройства, на которых проводится интеллектуальная обработка изображений, ограничены в потребляемой мощности. В свою очередь, потребляемая мощность вычислительного устройства коррелирует с производительностью устройства. Поэтому достижение высоких показателей качества распознавания и быстродействия вычислительно сложными СНС на устройствах с ограниченным вычислительным ресурсом является нетривиальной задачей.

Проблема 5. Построение СНС на устройствах с ограниченным вычислительным ресурсом требует новых подходов, которые позволят создавать распознающую интеллектуальную систему, удовлетворяющую необходимым характеристикам.

Требования к скорости обработки данных СНС в масштабе реального времени на устройствах с ограниченным вычислительным ресурсом привело к появлению новых аппаратных и программных решений. Особенностью вычислителей является аппаратная поддержка нейронного ускорителя. Вместе с вычислительным устройством предоставляется программное обеспечение (ПО), позволяющее воспользоваться вычислительными мощностями нейронного ускорителя. Существует большое разнообразие нейронных ускорителей, которые рассмотрены ниже.

Программируемая логическая интегральная схема (ПЛИС) или Field Programmable Gate Array (FPGA). FPGA позволяет реализовать СНС на аппаратном уровне.

В 2017 году компания Google представила тензорный процессор или TPU, который является специализированной интегральной схемой для ускорения нейросетевых вычислений. Тензорный процессор способен выполнять сотни тысяч операций за один такт.

Достоинства: TPU достигает высокой производительности, особенно для больших размеров входных данных.

Недостатки: программная реализация НС должна быть создана только с помощью библиотеки глубокого обучения *TensorFlow*. TPU оптимизированы для моделей глубокого обучения и могут быть не столь эффективны при выполнении других типов моделей машинного обучения.

Графический процессор или GPU – устройство персонального компьютера, выполняющее графический рендеринг. *Рендеринг* – это проектирование трёхмерной сцены в двухмерное изображение с помощью компьютерной программы с учётом заданных параметров: освещения, точки наблюдения, материалов объектов. Одним из лидеров в производстве графических процессоров является компания NVIDIA, которая разработала целый ряд библиотек, содержащих оптимизированные под графический процессор реализации базовых функций, необходимых для ускорения нейронных сетей: NVIDIA Graphics Processing Unit Computing Toolkit, Compute Unified Device Architecture (CUDA), CUDA Deep Neural Network (cuDNN), TensorRT и др. Библиотека CUDA – это библиотека, обеспечивающая набор инструментов для создания высокопроизводительных алгоритмов на графическом сопроцессоре. Библиотека cuDNN – это библиотека примитивов с ускорением на GPU для глубоких НС, которая оптимизирует производительность НС на графических процессорах NVIDIA. Библиотека TensorRT – это библиотека, которая позволяет обеспечить низкую временную задержку и высокую пропускную способность для алгоритмов обработки данных на графическом сопроцессоре.

Достоинства: GPU обладают высокой производительностью, поддержкой распараллеливания однотипных вычислений, поддержкой вычислений с различной разрядностью. Библиотека TensorRT от NVIDIA позволяет автоматизировать процесс квантования и оптимизации вычислительного графа модели НС. Разработка НС под графический процессор NVIDIA имеет преимущество из-за поддержки большого количества передовых библиотек глубокого обучения TensorFlow [100], PyTorch [101] и Keras [102]. Также компания NVIDIA предоставляет встраиваемые вычислительные платформы Jetson на базе искусственного интеллекта (ИИ).

Недостатки: высокая стоимость и потребляемая мощность.

Нейронный процессор или Neural Processing Unit (NPU) – это специализированный аппаратный ускоритель, предназначенный для обработки данных ИС. NPU обеспечивают высокую производительность при минимальном энергопотреблении, что делает их пригодными для устройств с ограниченным вычислительным ресурсом. NPU оптимизированы для операций, используемых в машинном обучении, таких как матричное умножение.

Достоинства: архитектура нейронного ускорителя NPU спроектирована так, чтобы обеспечить высокую энергоэффективность и производительность, что позволяет ускорить прямое распространение данных через ИС.

Недостатки: NPU не так широко доступны для общего пользования, как графические процессоры, поддерживают узкий спектр фреймворков глубокого обучения.

Обычно алгоритмам технического зрения приходится обрабатывать изображения высокого разрешения различных спектральных диапазонов, включая видимый, ИК, радиолокационный и другие диапазоны. Обработка таких изображений должна выполняться в масштабе реального времени, что требует специальных подходов построения ИС на устройствах с ограниченным вычислительным ресурсом.

Основываясь на вышесказанном, можно сформулировать актуальную научно–техническую задачу повышения качества распознавания малоразмерных изображений объектов ИС для устройств с ограниченным вычислительным ресурсом. Решение обозначенной научно-технической задачи существующими методами не представляется возможным из-за ряда основополагающих противоречий.

Противоречие в практике состоит в том, что разработка алгоритмов распознавания изображений объектов для устройств с ограниченным вычислительным ресурсом задаёт высокие требования к качеству распознавания изображениях и скорости обработки данных. С одной стороны, достижение высокого качества распознавания изображений объектов можно обеспечить вычислительно слож-

ными СНС. С другой стороны, распознавание объектов такими алгоритмами на устройствах с ограниченным вычислительным ресурсом не позволяет обрабатывать данные с необходимой скоростью.

Противоречие в теории обусловлено тем, что расширение признакового пространства новыми признаками не гарантирует формирование решающей функции НС, обладающей лучшей обобщающей способностью, из-за недостаточной различительности новых признаков.

1.11 Выводы по материалам первой главы

В настоящее время большое внимание уделяется алгоритмам цифровой обработки изображений на базе свёрточных нейронных сетей. Свёрточные нейронные сети востребованы в таких системах, как беспилотные автомобили, системы видеонаблюдения и др. Во избежание аварийных ситуаций распознающие системы должны иметь достаточно времени на реагирование для принятия дальнейших решений. Распознавание объектов на дальних расстояниях позволит повысить время на реагирование. Объекты на дальних расстояниях представлены малой группой пикселей, распознавание которых вызывает трудности даже у передовых архитектур свёрточных нейронных сетей.

Свёрточные нейронные сети строятся на вычислительных устройствах с ограниченным вычислительным ресурсом. Вычислительно сложным нейронным сетям приходится обрабатывать большие объёмы данных в масштабе реального времени. Этот факт накладывает ограничение на выбор аппаратной платформы, а также вынуждает применять специальные алгоритмические и программные решения для повышения скорости обработки данных алгоритмом. Таким образом, проведённый анализ от разработки алгоритмов до их построения позволил выявить главные проблемы:

1. Неудовлетворительное качество распознавания малоразмерных изображений свёрточными нейронными сетями обусловлено недостаточной инфор-

мативностью признакового пространства. Сокращение ошибок распознавания требует новых методов, позволяющих получить систему признаков с большим количеством различительных признаков.

2. Повышение качества распознавания малоразмерных изображений объектов свёрточных нейронных сетей требует новых алгоритмов предварительной обработки изображений, которые будут учитывать признаки движения, сформированные в процессе продолжительного наблюдения за объектами.

3. Классические подходы формирования датасетов разномодальных изображений приводят к неприемлемому повышению стоимости разработки распознающей системы на этапе обучения алгоритма, что требует новых эффективных подходов, позволяющих сократить затраты на разработку.

4. Обработка больших объёмов данных алгоритмами предварительной обработки изображений на устройствах с ограниченным вычислительным ресурсом может приводить к неприемлемой временной задержке выполнения алгоритма, что требует эффективных алгоритмических и программных решений, которые ускорят обработку данных.

5. Создание интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом, обеспечивающей необходимые характеристики в качестве распознавания, скорости обработки данных и допустимой потребляемой мощности вычислительного устройства, является нетривиальной задачей, которая требует эффективных решений.

Глава 2. РАСПОЗНАВАНИЕ МАЛОРАЗМЕРНЫХ ИЗОБРАЖЕНИЙ

2.1 Разработка метода расширения признакового пространства с целью повышения информативности системы признаков

Существует большое количество моделей представления образов для задачи распознавания. Большая часть моделей основывается на геометрической интерпретации представления образа, согласно которой каждый объект материального мира может быть представлен в виде точки A в n -мерном пространстве признаков. Декартовы координаты точки A являются признаками объекта x_1, x_2, \dots, x_n . Любая совокупность точек A_i может быть представлена в виде матрицы координат X . На рисунке 2.1 представлено геометрическое представление объекта в виде точки в трёхмерном пространстве признаков.

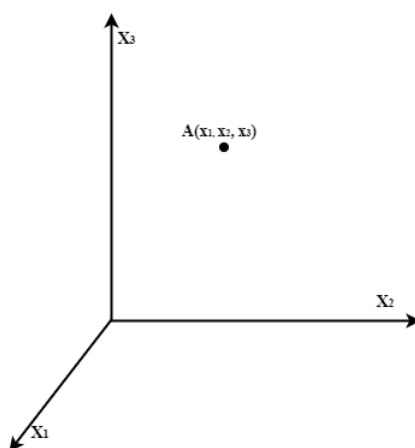


Рисунок 2.1 – Геометрическое представление объекта в виде точки в трехмерной системе признаков

Множество точек $A_i (i = 1, \dots, p)$ может быть ограничено некоторой многомерной гиперплоскостью. Если провести описание объектов двух образов, то каждому образу будет соответствовать область в выбранном пространстве признаков (рисунок 2.2). Если области B_1 и B_2 не пересекаются друг с другом, то задача распознавания сводится к задаче различения объектов. Это позволяет сделать однозначный вывод о принадлежности объекта к конкретному образу. В

противном случае, когда области пересекаются, появляются зоны неоднозначности и задача сводится к классификации объектов. Наличие зон неоднозначности обуславливает ошибки классификации объектов, что можно объяснить неполным признаковым описанием объектов. Это связано с тем, что какие-то характеристики наблюдаемого объекта не удаётся использовать для распознавания.

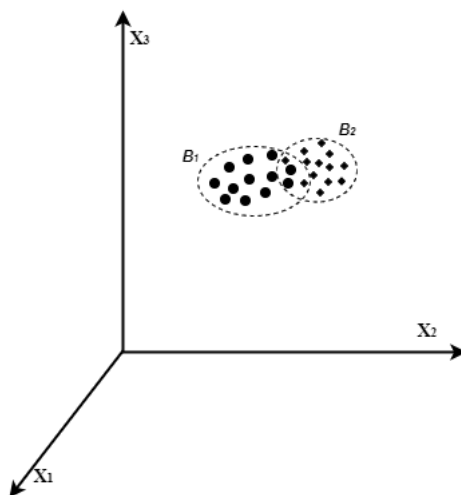


Рисунок 2.2 – Отображение множества объектов в виде точек в трехмерной системе признаков, где B_1 , B_2 – области двух образов в трёхмерной системе признаков

2.2 Постановка задачи

Для решения первой проблемы диссертационной работы, сформулируем постановку задачи. Задана обучающая $Z_{об}$ и тестовая $Z_{тест}$ выборки изображений, которые определяют систему признаков $X = \{x_i\}$. В системе признаков X определены границы целевых образов $S = \{s_i\}$ с помощью описания эталонов (разметка, аннотирование), допустимая величина затраты N_0 . Задан тип решающей функции D в виде архитектуры СНС. Решающая функция классификатора D строится по заданной обучающей выборке $Z_{об}$ с помощью обучаемых итеративных алгоритмов (метод обратного распространения ошибки). Задача обучения заключается в минимизации ошибок распознавания R на тестовой выборке путём определения весовых коэффициентов $W = \{w_i\}$ модели СНС.

Требуется получить более информативную систему признаков $Y = \{y_i\}$ из исходной системы признаков X , использование которой позволит сформировать

весовые коэффициенты W , определяющие разделяющую поверхность решающей функции D с меньшей ошибкой распознавания на тестовой выборке при минимальных затратах на получение нового пространства признаков $N(Y)$.

Решение поставленной задачи осуществляется СНС, построение которых основано на модели искусственного нейрона или перцептрона. Входными данными перцептрона являются значения x_1, x_1, \dots, x_n (рисунок 2.3), где n – количество входов. Каждое поступающее значение умножается на соответствующий ему вес w_i , затем вычисляется взвешенная сумма входов и к ней прибавляется свободный член смещения b (bias). В результате формируется величина net , которая преобразуется в функции активации σ в результирующее значение Out :

$$Out = \sigma \sum (x_i w_i + b). \quad (2.1)$$

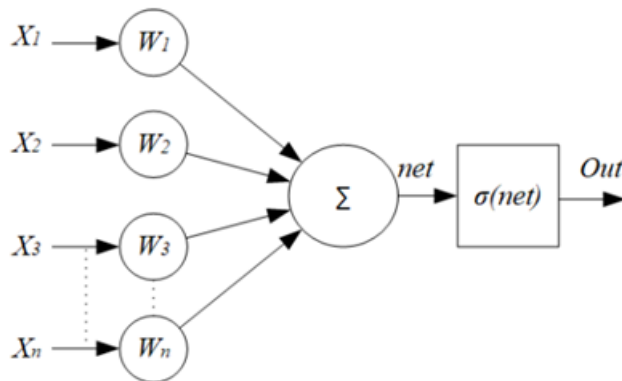


Рисунок 2.3 – Схема искусственного нейрона

Наличие функции активации σ имеет большое значение в архитектуре НС, так как именно она вносит нелинейность в систему и выявляет сложные зависимости в данных. Рассмотрим модель многослойной НС с полносвязными слоями, обеспечивающую отнесение объекта к одному из двух заданных классов. Архитектура многослойной НС состоит из одного входного слоя, скрытых слоев и выходного слоя (рисунок 2.4).

Каждый нейрон многослойного перцептрона полностью связан со всеми нейронами предшествующего слоя. Реакция всей системы S пропорциональна сумме взятых с определёнными весами реакций последнего скрытого слоя. Для

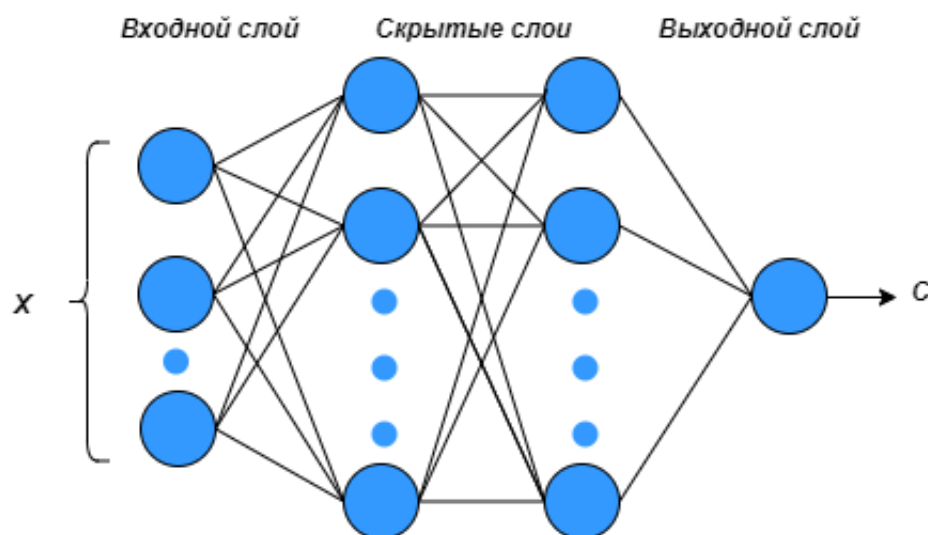


Рисунок 2.4 – Архитектура многослойной нейронной сети с полносвязными слоями

случаев, когда $C > 0$, принимается решение о принадлежности к классу B_1 , в противном случае – к классу B_2 .

2.3 Метод расширения признакового пространства с целью повышения информативности системы признаков

Решение первой проблемы. В большинстве случаев распознающим системам достаточно оценить набор признаков для одного момента времени, чтобы с высокой степенью вероятности распознать объект. Однако в сложных ситуациях количественное описание свойств объекта для одного момента времени бывает недостаточно. Поэтому приходится анализировать историю изменений свойств объекта. Для того чтобы организовать процедуру распознавания с учётом предыстории, необходимо чтобы распознающая система принимала решение на основе признаков, сформированных в процессе продолжительного наблюдения. С учётом данной концепции предложен метод расширения признакового пространства с целью повышения информативности системы признаков.

Предложенный метод опирается на априорные знания о характере признаков объектов, протекающих во времени. Делается предположение, что объекты подвижны, изображения имеют малые размеры и претерпевают геометрические и яркостные изменения. Тогда парадигма метода заключается в формировании

нового признакового пространства Y путём преобразования исходных данных, формирующего признаковое пространство X , в котором минимизированы зоны неоднозначности посредством увеличения расстояния между границами образов [103, 104]. Это достигается за счет расширения исходного признакового пространства новыми различительными признаками движения.

Формирование решающей функции D модели СНС осуществляется с помощью обучения на тренировочной выборке, которая определяет признаковое пространство Y . В процессе обучения СНС вычисляются весовые коэффициенты искомого решения, которые определяют разделяющую гиперплоскость. Решающая функция D , полученная на данных с расширенным признаковым пространством, формирует весовые коэффициенты, которые определяют границу между образами для классификации объектов с меньшим количеством ошибок.

Стоит отметить, что повышение размерности признакового пространства влечёт за собой увеличения количества нейронов (фильтров) входного слоя СНС, что приводит к незначительному увеличению сложности СНС.

Для пояснения концепции предлагаемого метода приведём пример отображения объектов двух классов в двухмерном и трёхмерном признаковом пространстве (рисунок 2.5).



Рисунок 2.5 – Отображение множества объектов в виде точек в двухмерном и трёхмерном признаковом пространстве

В двухмерном признаковом пространстве образы двух классов имеют зоны неоднозначности, т.е. пересекаются [103]. Обучение НС на тренировочной выборке с исходным признаковым пространством приводит к построению решающей функции $D(X)$. В результате чего не для всех объектов, принадлежащих образу B_1 , выполняется неравенство $D(x_1, x_2) < 0$ и объектов образа B_2 выполняется $D(x_1, x_2) > 0$. Обучение СНС на данных, представляющих двухмерное признаковое пространство, определили весовые коэффициенты модели с ошибками классификации. При этом добавление нового различительного признака приводит к формированию решающей функции $D(Y)$ в трёхмерном признаковом пространстве, которая обладает свойствами: для всех объектов образа B_1 выполняется неравенство $D(y_1, y_2, y_3) < 0$, и для всех объектов образа B_2 выполняется неравенство $D(y_1, y_2, y_3) > 0$, что означает безошибочную классификацию. Обучение НС на данных, которые представляют трёхмерное признаковое пространство, приводит к формированию весовых коэффициентов, которые определили разделяющую границу классификатора, однозначно разграничивающую образы двух классов в пространстве признаков, что позволяет предотвратить переобучение. Таким образом, если задано множество объектов двух классов B_1 и B_2 в расширенном трёхмерном признаковом пространстве Y , то решение сводится к отысканию вектора W в пространстве Y , обладающего справедливостью для $D(Y)$:

$$D(y_1, y_2, y_3|_{B_1}) < 0, \quad (2.2)$$

$$D(y_1, y_2, y_3|_{B_2}) > 0. \quad (2.3)$$

На рисунке 2.6 изображена структурная схема предварительной обработки предлагаемого метода. Сенсорные датчики распознающей системы фиксируют изображения визуального представления мира с определенным временным интервалом. По характеру описания временных процессов, происходящих в последовательности изображений, признаки можно разделить на статические и динамические [1]. Статические признаки описывают лишь пространственные

свойства изображения, а динамические признаки описывают свойства объектов во времени. Динамические признаки можно получить только на основе последовательности изображений или видеоизображений. В процессе движения объекта группа пикселей претерпевает геометрические, яркостные и пространственные изменения для каждого момента времени.

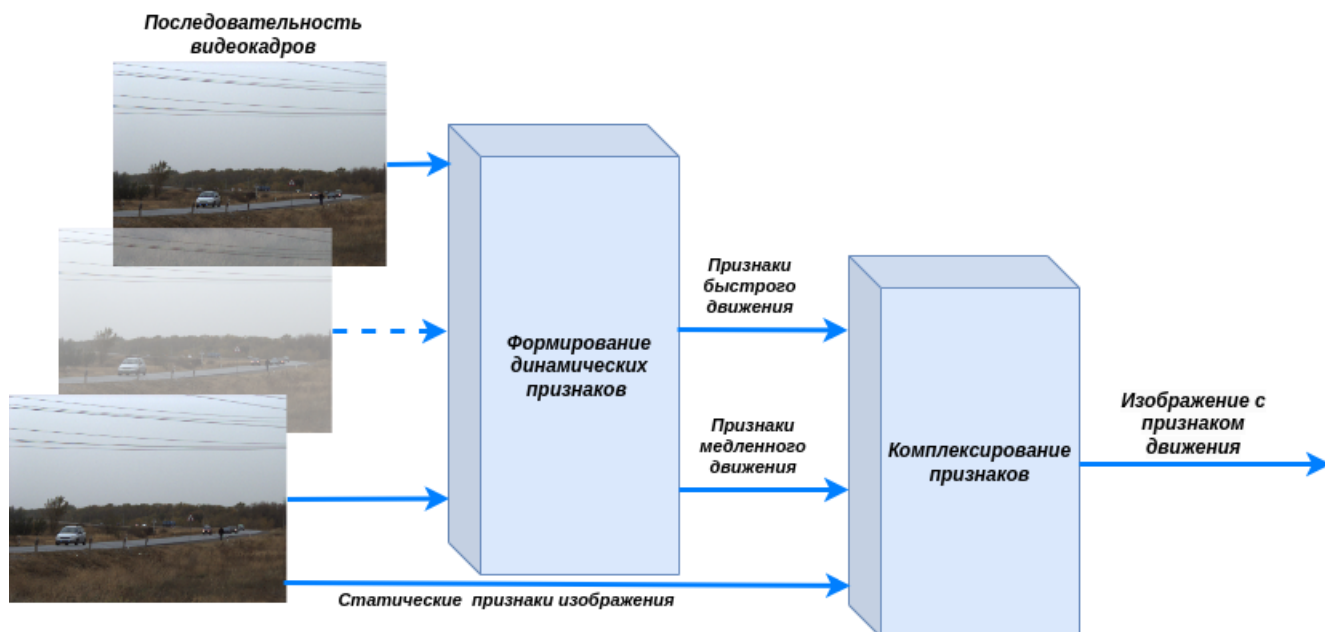


Рисунок 2.6 – Структурная схема предварительной обработки метода расширения признакового пространства с целью увеличения информативности системы признаков

На первом этапе в блок формирования динамических признаков поступает временная последовательность изображений, на основе которой выделяются признаки движения, характеризующие различную скорость движения объектов $Q = \{q_i\}$. В результате чего формируются новые признаки, характеризующие движение объектов в процессе их изменения [105]. Выделение признаков движения основано на анализе временной близости пикселей в последовательности кадров.

На втором этапе происходит формирование результирующих изображений, которые определяют новое признаковое пространство Y с помощью объединения статических признаков X со всей совокупностью признаков движения Q . Таким образом, результирующее изображение содержит исходные признаки изображения и новые признаки движения.

Достоинства разработанного метода: 1) в новом признаковом пространстве учтены исходные признаки изображения и добавлены новые динамические признаки движения, что приводит к повышению качества распознавания малоразмерных изображений подвижных объектов при сохранении качества распознавания малоразмерных изображений неподвижных объектов; 2) применим к широкому спектру архитектур СНС. Далее рассмотрены процессы обучения и распознавания СНС с применением разработанного метода расширения признакового пространства.

2.4 Обучение и распознавание с применением разработанного метода расширения признакового пространства

Условно процесс обучения СНС распознаванию малоразмерных изображений объектов с расширенным признаковым пространством можно представить в виде структурной схемы на рисунке 2.7.

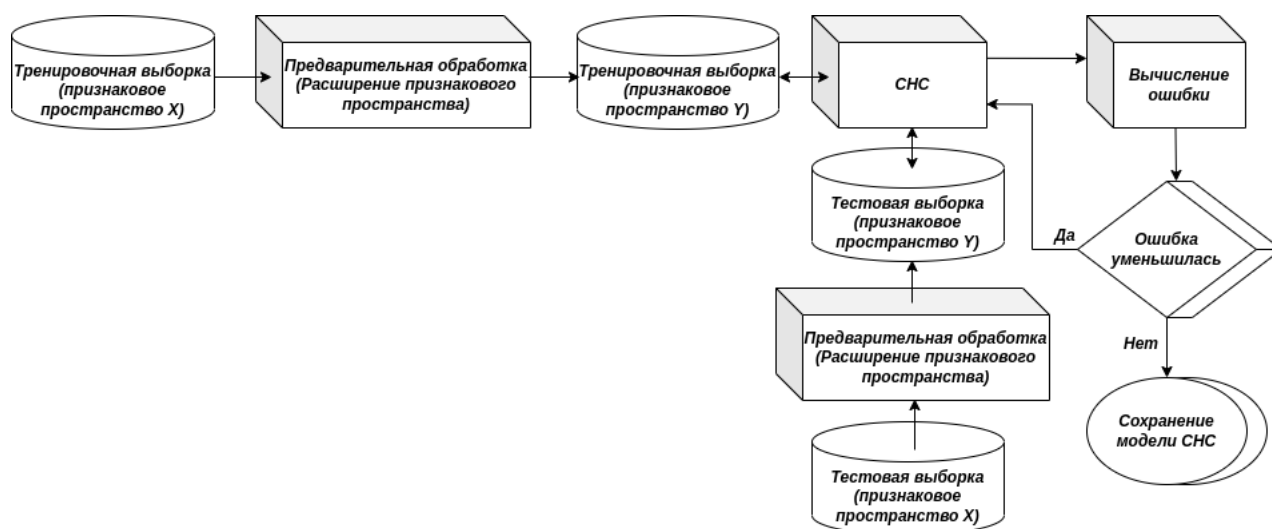


Рисунок 2.7 – Структурная схема обучения сверточных нейронных сетей с применением разработанного метода расширения признакового пространства

Алгоритм обучения СНС не меняется. Изменяется только тренировочная и тестовая выборки. Рассмотрим пошаговое обучение СНС с применением предложенного метода.

1. Формируются обучающая и тестовая выборки аннотированных изображений.

2. На основе созданных выборок формируются новые выборки аннотированных изображений с помощью предварительной обработки изображений.

3. Из тренировочной выборки аннотированных изображений берётся партия образцов (батчей) изображений, которые представляют тензор (многомерный массив чисел) с эталонной информацией о метках классов и расположении объектов на изображениях.

4. Далее батч поступает на вход СНС и прямо распространяется до формирования решения.

5. Предсказанное решение алгоритма сравнивается с эталонными образцами, формируя значение ошибки по заданной формуле.

6. Полученная ошибка распространяется обратно по СНС методом Backpropagation [43], вычисляя и применяя поправки к весовым коэффициентам модели.

7. После обработки всех образцов изображений в тренировочной выборке рассчитывается среднее значение ошибки на тестовой выборке. Если ошибка меньше, чем на предыдущей итерации обучения, то происходит переход к шагу 3 (процесс обучения повторяется). В противном случае процесс обучения СНС завершается, и модель СНС сохраняется.

Процесс распознавания обученной модели СНС с применением разработанного метода расширения признакового пространства представлен на рисунке 2.8.

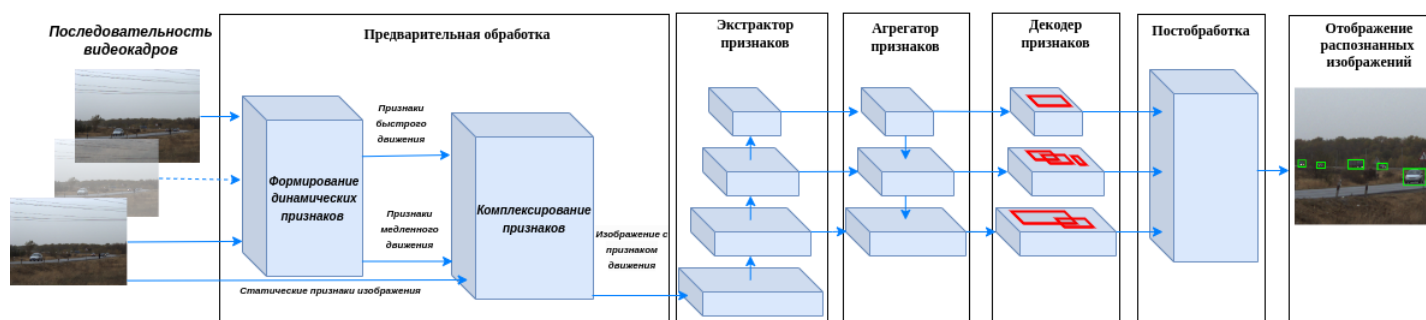


Рисунок 2.8 – Распознавание свёрточными нейронными сетями с применением предварительной обработки метода расширения признакового пространства

Основная концепция распознавания малоразмерных изображений с применением предложенного метода заключается в обработке изображений, пред-

ставляющих признаков пространство Y . Для этого необходимо формировать изображение, в котором учтены признаки движения с помощью предварительной обработки последовательности изображений. Применение такого подхода увеличивает затраты на измерение $N(Y)$ и обработку признаков $N(D)$ в виде увеличения временной задержки и потребляемого вычислительного ресурса. Увеличение затрат $N(D)$ связано с повышением количества обрабатываемых признаков входным слоем СНС, что приводит к незначительному увеличению количества фильтров входного слоя СНС.

2.5 Распознавание малоразмерных изображений свёрточными нейронными сетями

К бортовым системам видеоаналитики предъявляются высокие требования к качеству распознавания объектов и скорости обработки данных, которые обеспечиваются одноэтапными СНС. Среди одноэтапных архитектур СНС широкое применение имеют: SSD, YOLO, CenterNet, FCOS и др. Рассмотрим перспективные архитектуры одноэтапных СНС, которые целесообразно применять для распознавания малоразмерных изображений.

You Only Look Once V3 – это архитектура СНС прямого распространения, которая была представлена Joseph Redmon [14] в 2016 году. Базовым экстрактором признаков YOLOv3 является Darknet53, который включает 53 операции «двухмерная свёртка» (Convolution 2D). Однако в качестве экстрактора признаков могут успешно применяться и другие архитектуры: Darknet19, ResNet, RetinaNet и др.

При разработке бортовых систем, где есть ограничения по времени выполнения алгоритма, целесообразно применять архитектуры СНС, основанные на двухмерной поканальной свёртке. К числу таких архитектур можно отнести MobileNetV2, MobileNetV3 и др.

Агрегация семантической информации с различных масштабов пирамиды признаков осуществляется методом FPN. Декодирование агрегированных призна-

ков осуществляется одноэтапным методом YOLO (рисунок 2.9). Глубина тензора на выходе агрегатора признаков:

$$d = b(t_x + t_y + t_w + t_h + p_o + c), \quad (2.4)$$

где t_x и t_y – значения, характеризующие смещение центра обрамляющего прямоугольника; t_w и t_h – значения, характеризующие изменение ширины и высоты обрамляющего прямоугольника; p_o – значение, характеризующее уверенность объект/не объект; $c = (p_1, p_2, \dots, p_c)$ – вектор уверенностей по классам p ; b – количество опорных прямоугольников для данного выхода с пирамиды признаков.

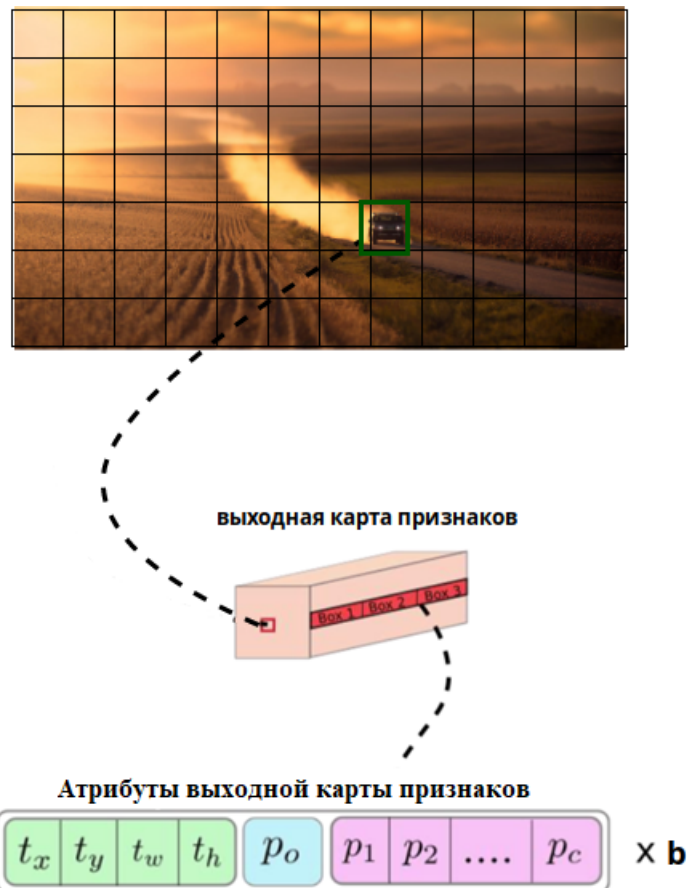


Рисунок 2.9 – Декодирование выходной карты признаков методом YOLO

Метод YOLO использует априорную информацию о размерах изображений объектов в тренировочной выборке. На основе априорной информации рассчитываются значения ширины и высоты опорных прямоугольников, необходимых для алгоритма YOLO.

Постобработка предсказанного множества ограничивающих рамок осуществляется алгоритмом NMS, который формирует результирующее множество решений.

You Only Look Once V4 представлена Alexey Bochkovskiy, Chien-Yao Wang и Hong-Yuan Mark Liao в 2020 году [7, 106, 107]. Базовыми архитектурами YOLOv4 являются YOLOv4-CSP, YOLOv4-tiny и YOLOv4-large [74]. В качестве экстрактора признаков YOLOv4-CSP использует CSPDarknet53, а YOLOv4-tiny использует CSPOSANet. YOLOv4-large включает вычислительно сложные архитектуры различной глубины YOLOv4-P5, YOLOv4-P5, YOLOv4-P7, которые предназначены для достижения высоких показателей качества распознавания. Архитектуры YOLOv4-large созданы для обучения на облачных графических процессорах из-за своей вычислительной сложности. Экстракторы признаков используют подход (CSP)(Cross-Stage-Partial-Connections) для снижения вычислительно сложных операций, связанных с дублированием информации о градиентах без изменений качества распознавания. В YOLOv4 добавлен блок Spatial Pyramid Pooling (SPP) [75], который увеличивает рецептивное поле, выделяет наиболее значимые особенности контекстной информации с минимальными затратами. В YOLOv4 агрегирование признаков осуществляется методом PAN. После чего признаки декодируются методом YOLO.

Также в YOLOv4 используются новые решения: аугментация данных Mosaic, перекрестная мини-пакетная нормализация Cross mini-Batch Normalization (CmBN), регуляризация DropBlock, функция ошибки Complete Intersection over Union (CIoU), самосопротивляющееся обучение Self-Adversarial-Training (SAT), функция активации Mish [108]. Распространённым фреймворком обучения моделей YOLOv3 и YOLOv4 является Darknet [109, 110].

You Only Look Once V5. Архитектура YOLOv5 выпущена компанией Ultralytics. YOLOv5 объединила в себе последние инновации, аналогичные архитектуре YOLOv4. YOLOv5 реализована на одном из самых популярных фреймворков в области глубокого обучения *Pytorch*, написанным на языке программирования *Python*. Авторы YOLOv5 не опубликовали подробную ста-

тью, а лишь предоставили программные компоненты в открытый доступ [111]. На основе программных компонент выпущены научные статьи применения и разбора архитектуры YOLOv5 от других авторов [112, 113]. В основе архитектуры YOLOv5 лежит экстрактор признаков типа CSP, агрегирование признаков осуществляется методами SPP и PANet, декодирование методом YOLOv3 с использованием функции потерь Generalized Intersection over Union (GIoU). К числу базовых архитектур YOLOv5 относятся: YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x. Модели сформированы на основе базовой архитектуры путем масштабирования по глубине и ширине. Программная реализация YOLOv5 рассчитывает опорные прямоугольники для декодирования признаков на этапе обучения в отличие от фреймворка Darknet.

You Only Look Once V8. Алгоритм YOLOv8 [114, 115] был выпущен в 2023 году компанией Ultralytics, которая разработала YOLOv5. Ultralytics предоставили пять архитектур СНС с различной сложностью: YOLOv8n (nano), YOLOv8s (small), YOLOv8m (medium), YOLOv8l (large) и YOLOv8x (extra-large). В YOLOv8 используются высокоуровневые признаки с контекстной информацией для повышения показателей качества распознавания. В YOLOv8 используется та же архитектура, что в YOLOv5 с некоторыми изменениями в CSP модуле, который теперь называется Convolutional to Feature. YOLOv8 использует модель без привязки к опорным прямоугольникам, новую функции потерь CIoU [116] и DFL [117], что позволило повысить характеристику качества распознавания mAP на открытом датасете COCO.

Detection Transformer. Появление первого алгоритма на базе трансформеров [118] постепенно распространилось на большинство задач машинного обучения от обработки естественного языка до компьютерного зрения, где распознавание объектов по их изображениям не стало исключением. Архитектуры СНС на базе трансформеров основаны по принципу кодер-декодер. Одним из современных алгоритмов на базе трансформеров является алгоритм DETR [72]. DETR не нуждается в применении постобработки и не использует априорную информацию о размерах изображений объектов. В настоящее время алгоритмы

распознавания изображений на базе трансформеров показывают одни из лучших результатов на соревновательных датасетах. Однако у трансформеров есть существенный недостаток, который заключается в сложности обучения модели. Обучение трансформеров требует большого вычислительного ресурса и тонкой настройки гиперпараметров модели для успешной сходимости алгоритма.

Распознавания малоразмерных изображений вызывает трудности даже у современных СНС, таких как YOLOv5, YOLOv8, DETR. Это связано с проблемой переобучения модели СНС из-за малого количества различительных признаков для изображений малых размеров. Вычислительные сложности современных архитектур СНС продолжают расти в угоду минимизации ошибки распознавания, что может приводить к ускорению переобучения СНС в задаче распознавания малоразмерных изображений. Поэтому для предотвращения переобучения моделей СНС предложено расширить признаковое пространство новыми независимыми признаками движения объектов, сформированными в процессе их изменения. Расширение исходного признакового пространства основывается на предварительной обработке последовательности видеок кадров. Далее рассмотрен алгоритм, реализующий предварительную обработку метода.

2.6 Разработка алгоритма предварительной обработки изображений с целью комплексирования признаков движения

В разработанном методе расширение признакового пространства с целью повышения информативности системы признаков осуществляется с помощью применения предварительной обработки последовательности изображений. Задача предварительной обработки заключается в выделении динамических признаков из последовательности изображений и комплексировании со статическими признаками изображения. Формирование динамических признаков осуществляется с помощью алгоритмов цифровой обработки изображений, которые учитывают временную близость процессов, протекающих во времени. К числу таких алгоритмов можно отнести: разностный, статистический, оптический поток Лу-

каса – Канаде [119] и др. Также алгоритм идентификации движения должен учитывать последовательность изображений большей длины для формирования признаков движения объектов, характеризующих различную скорость. Главной задачей при построении алгоритма идентификации движения является формирование изображений с признаками, характеризующими объекты в процессе изменения.

Сформированные признаки движения необходимо скомплексировать с исходными признаками изображений. Так, в работах [8, 9] авторы комплексируют исходные признаки изображений методом главных компонент и дискретным вейвлет преобразованием. Недостатком этих подходов являются большие вычислительные затраты. Другим методом комплексирования данных являются поканальное объединение изображений на уровне пикселей [22]. Такой подход формирует изображение, у которого количество каналов равно количеству входных изображений. Достоинством этого подхода является простота реализации и низкая вычислительная сложность.

Решение второй проблемы. Основная концепция алгоритма заключается в формировании комплексированного изображения, в котором учтены признаки движения объектов в процессе их развития на основе пространственно-временной обработки последовательности изображений [92, 120–124].

Подразумевается, что на вход алгоритма поступают последовательные во времени кадры в градациях серого со статичным фоном в формате 8 бит. На первом этапе последовательные во времени изображения накапливаются в линии задержки с отводами (рисунок 2.10). Каждый видеокادر необходимо преобразовать из целочисленного типа данных к типу данных с плавающей точкой *float32* и провести нормализацию. Это сделано для того, чтобы избежать переполнения типа данных в дальнейших преобразованиях. Период выдачи кадров камерой определяется временной задержкой T_0 . Накопленная последовательность видеокладов поступает в блок идентификации движения.

С учётом компромисса между качеством идентификации движения и вычислительной сложностью, алгоритм идентификации движения основан на постро-

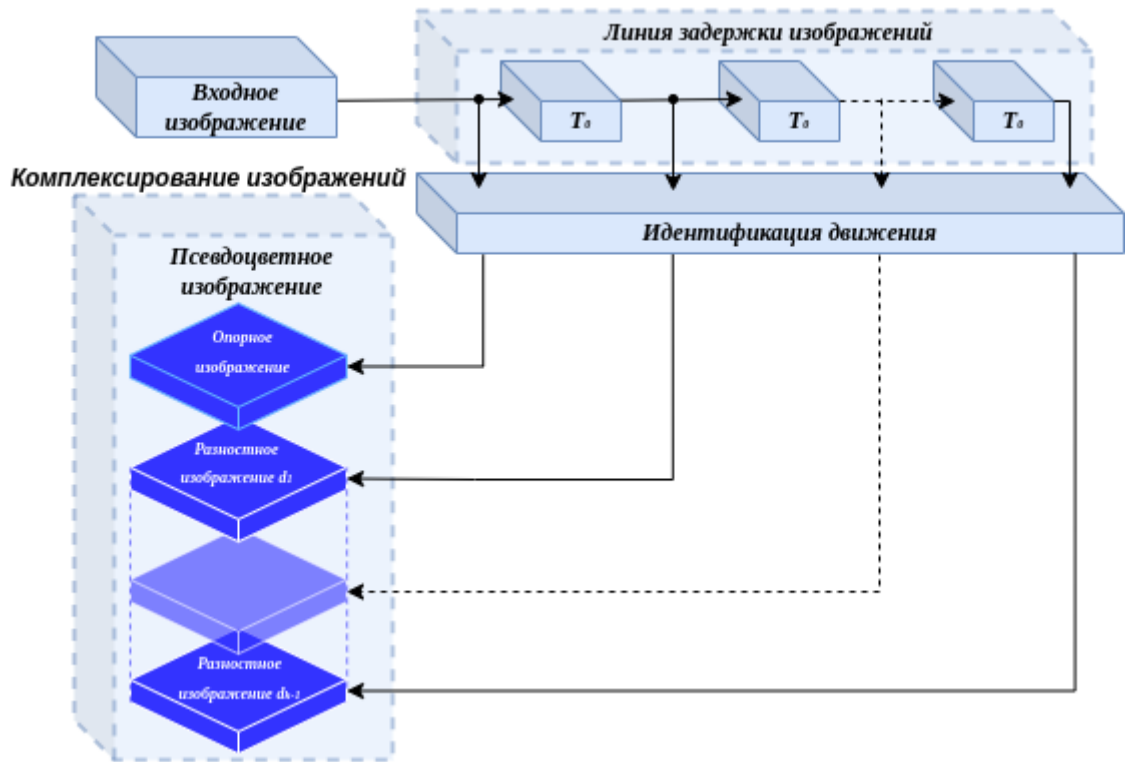


Рисунок 2.10 – Структурная схема алгоритма предварительной обработки изображений с целью комплексирования признаков движения

ении разностных изображений между опорным изображением и задержанными изображениями (рисунок 2.11):

$$d_i(x,y) = |f(x,y,t_0) - f(x,y,t_{i+1})|, \quad (2.5)$$

где i – индекс разностного изображения, который находится в диапазоне от 0 до $k - 1$, где k – количество задержанных кадров; $f(x,y,t_0)$ и $f(x,y,t_{i+1})$ – изображения, полученные в моменты времени t_0, t_{i+1} ; x и y – пространственные координаты. Изображение, полученное в момент времени t_0 , является опорным, а изображения, полученные в моменты времени t_i , где $i = 1, 2, \dots, k - 1$, являются задержанными. Разностные изображения d_i формируются путём сравнения опорного изображения с задержанными изображениями. Таким образом, разностное изображение, полученное в начальные моменты времени, отображает наличие медленного движения в видеокадре, а в поздние моменты времени отображает наличие быстрого движения. Взятие абсолютных значений разностных изображений

жений позволяет привести значения яркостей к одному динамическому диапазону от нуля до единицы во всех каналах комплексированного изображения.

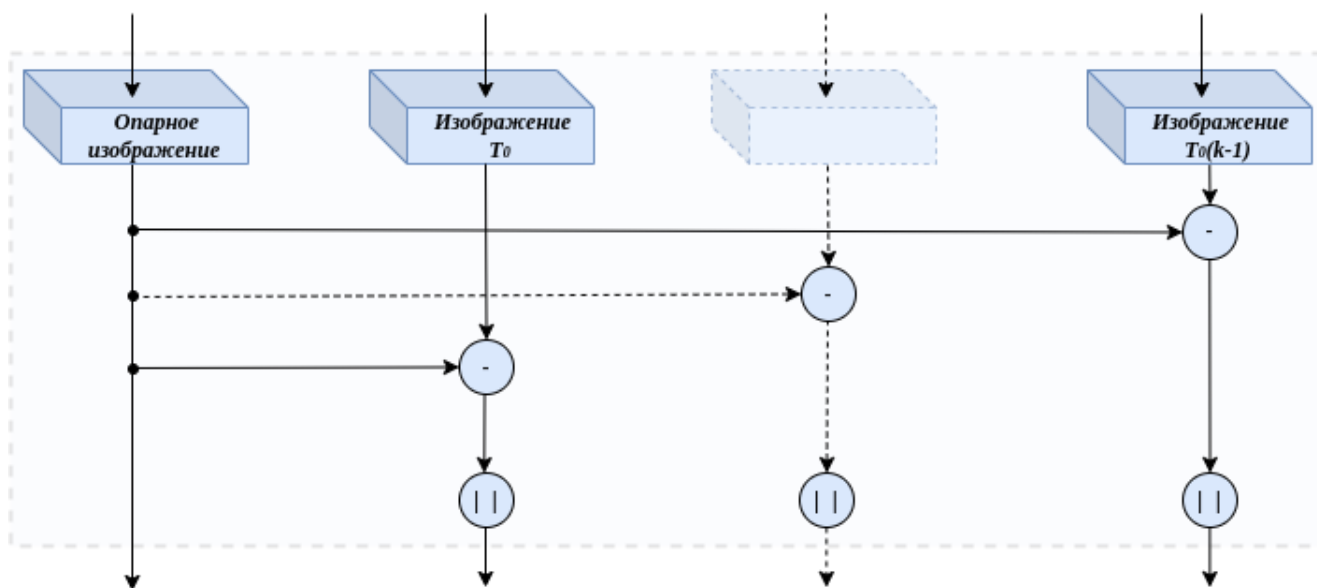


Рисунок 2.11 – Структурная схема алгоритма идентификации движения

Результирующее псевдоцветное изображение формируется на основе поканального объединения на уровне пикселей опорного изображения и разностных изображений для заданных моментов времени. Количество каналов разностных изображений не фиксировано и задаётся разработчиком. В приложении A.1 представлены программные компоненты, реализующие разработанный алгоритм на языке программирования *Python*.

Конечным результатом разработанного алгоритма предварительной обработки изображений является псевдоцветное изображение, характерное «цветными» локальными областями, означающими признаки движения. На рисунках 2.12, 2.13 приведены исходное и комплексированные изображения с одним и двумя разностными каналами в ИК и видимом оптических диапазонах.

Подразумевается, что алгоритм предварительной обработки последовательности видеок кадров использует изображения со статичным фоном. Если камера подвижна, то последовательность изображений можно заставить стабилизировать специальными алгоритмами, которые учитывают межкадровое смещение. Разработанный алгоритм отличается от известных тем, что расширяет исходное признаковое

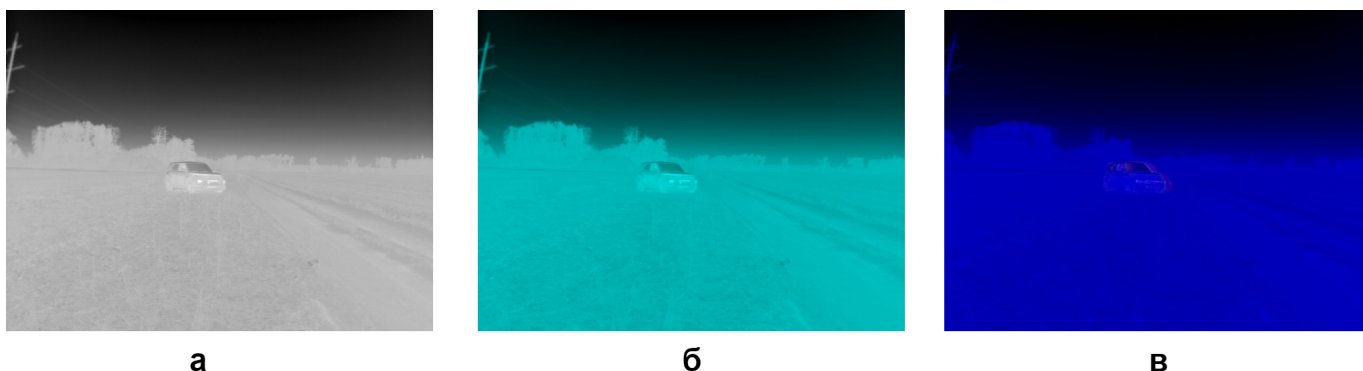


Рисунок 2.12 – Типовые изображения инфракрасного оптического диапазона: а) исходное изображение; б) комплексированное изображение с одним разностным каналом; в) комплексированное изображение с двумя разностными каналами

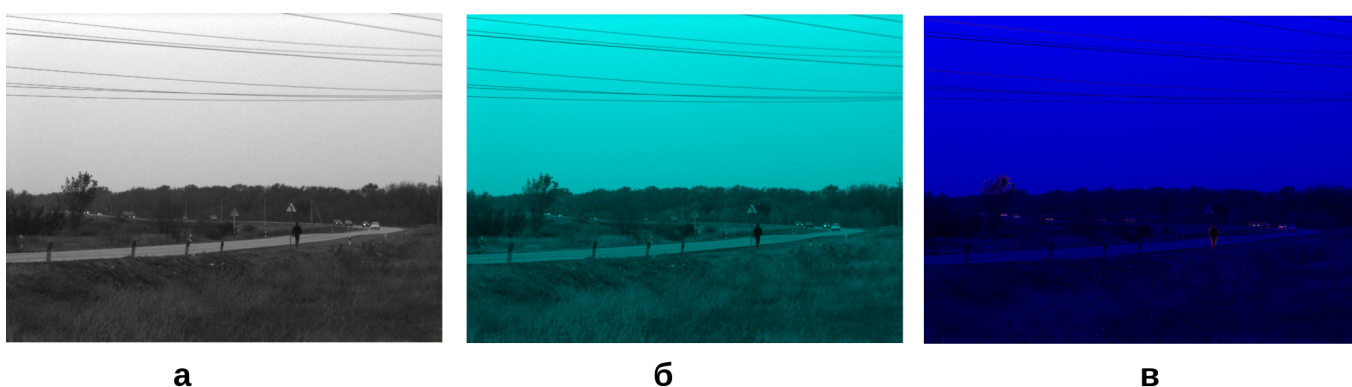


Рисунок 2.13 – Типовые изображения видимого оптического диапазона: а) исходное изображение; б) комплексированное изображение с одним разностным каналом; в) комплексированное изображение с двумя разностными каналами

пространство признаками движения объектов, сформированными в процессе длительного наблюдения.

2.7 Разработка алгоритма формирования аннотированных условно-реальных разномодальных изображений

Решение третьей проблемы. С учётом выявленной проблемы, связанной с повышением стоимости разработки распознающей системы из-за увеличения затрат на формирование датасетов, предлагается новое решение, основанное на автоматической генерации аннотированных синтетических изображений с помощью среды разработки 3D-графики Unreal Engine 5. Использование Unreal Engine

5 позволяет формировать реалистичные синтетические видеосюжеты с широким спектром различных моделей и текстур. Для обучения СНС, кроме данных изображений, нужна информация о индексе классов и расположении целевых изображений объектов на всём изображении. Эта информация называется разметкой и содержится в файлах аннотаций. Такой подход позволяет автоматизировать формирование разметки без ошибок, свойственных ручному способу.

Алгоритм формирования комплексированных разномодальных датасетов изображений (расширенное признаковое пространство) представлен в виде структурной схемы на рисунке 2.14.

На первом этапе необходимо смоделировать сценарий: выбрать карту местности, расположить объекты интереса в поле зрения видеокамеры, задать траектории движения объектов и др. Для удовлетворения критерия малоразмерности изображений необходимо расположить объекты на необходимом расстоянии от места их фиксации, так чтобы минимальный размер изображения удовлетворял необходимым вероятностям распознавания по критерию Джона Джонсона, которые определяются по формуле (1.4).

Реалистичность моделей и текстур достигается за счет анимации элементов фона и объектов от кадра к кадру: подвижностью травы, деревьев, облаков и др. Для каждого момента времени изображения объектов интереса претерпевают геометрические и яркостные изменения. Сканирование карты местности камерой является настраиваемым режимом. На рисунке 2.15 представлено изображение процесса моделирования карты местности с помощью среды разработки 3D-графики Unreal Engine 5.

Средство моделирования 3D-графики Unreal Engine 5 позволяет получить информацию о синтезируемой сцене. На рисунке 2.15 представлено цветное изображение видимого оптического диапазона, сформированное с помощью средства 3D-графики. При формировании цветного изображения видимого оптического диапазона используются вспомогательные каналы, такие как:

1. Карта глубины – это матрица, значения которой являются трёхмерными векторами или нормальными единичной длины к поверхности (рисунок 2.16 б).

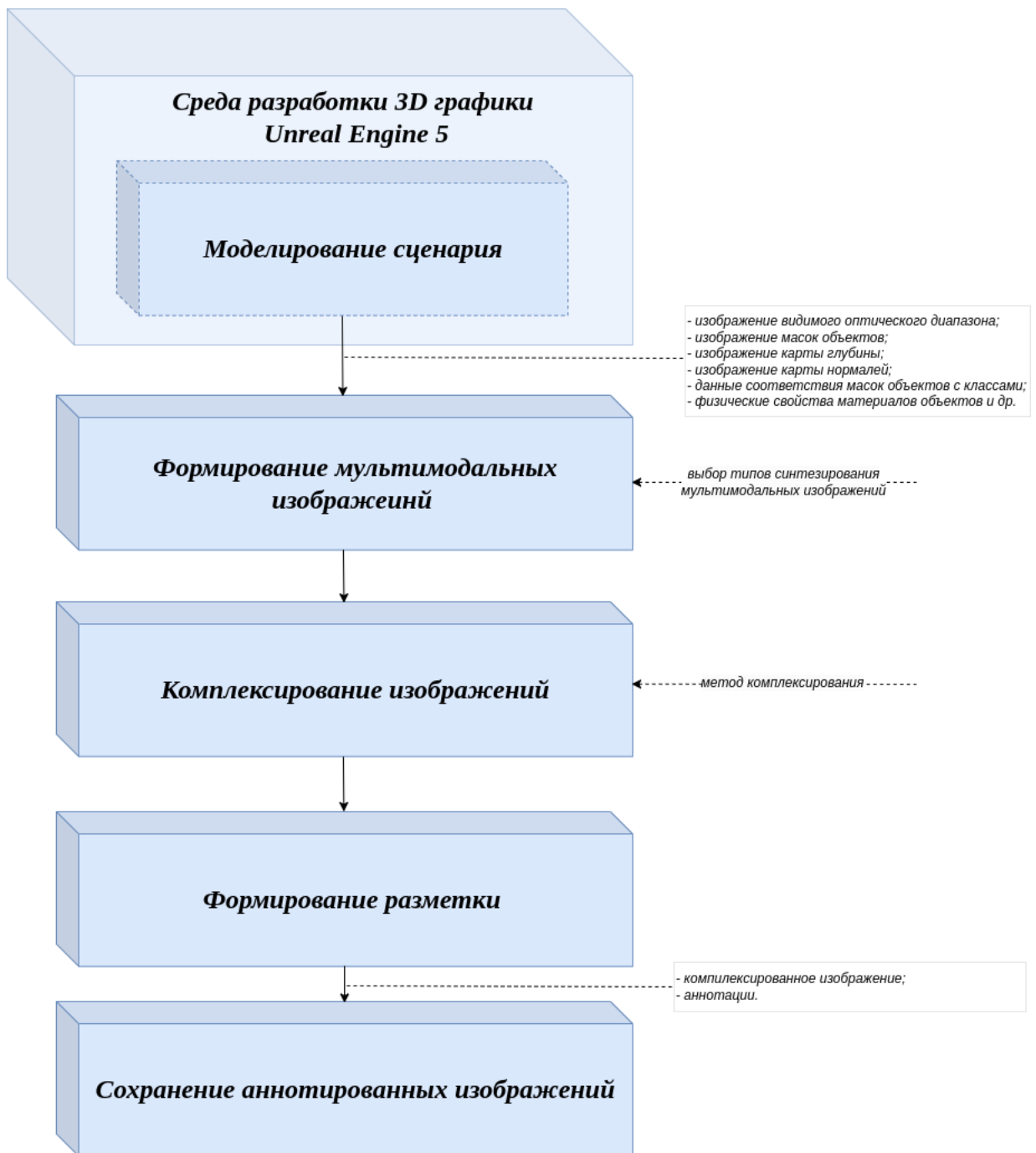


Рисунок 2.14 – Структурная схема алгоритма формирования аннотированных комплексированных разномодальных условно-реальных изображений

2. Карта нормалей – это матрица трёхмерных векторов нормалей к поверхностям (для удобства отображения карта нормалей изображена непосредственно в цветовом пространстве Red Green Blue (RGB)).

3. Карта масок – это матрица, значения которой содержат уникальные индексы объектов и поверхностей. Каждому индексу маски объектов и поверх-



Рисунок 2.15 – Моделирование сценария на основе Unreal Engine 5

ностей соответствует определенному классу. Такой подход позволяет различить объекты одного класса при перекрытии.



а



б

Рисунок 2.16 – Возможные реализации изображений, используемые при генерации: а) изображение видимого оптического диапазона; б) карта глубины

С учётом всех возможностей средств разработки 3D-графики существует возможность формирования разномодальных изображений. К числу таких изображений относятся синтетические ИК-изображения [125], радиолокационные изображения [126], изображения с признаками движения (рисунок 2.17) и др.

Далее необходимо провести комплексирование сформированных разномодальных изображений выбранным способом. К числу наиболее распростра-

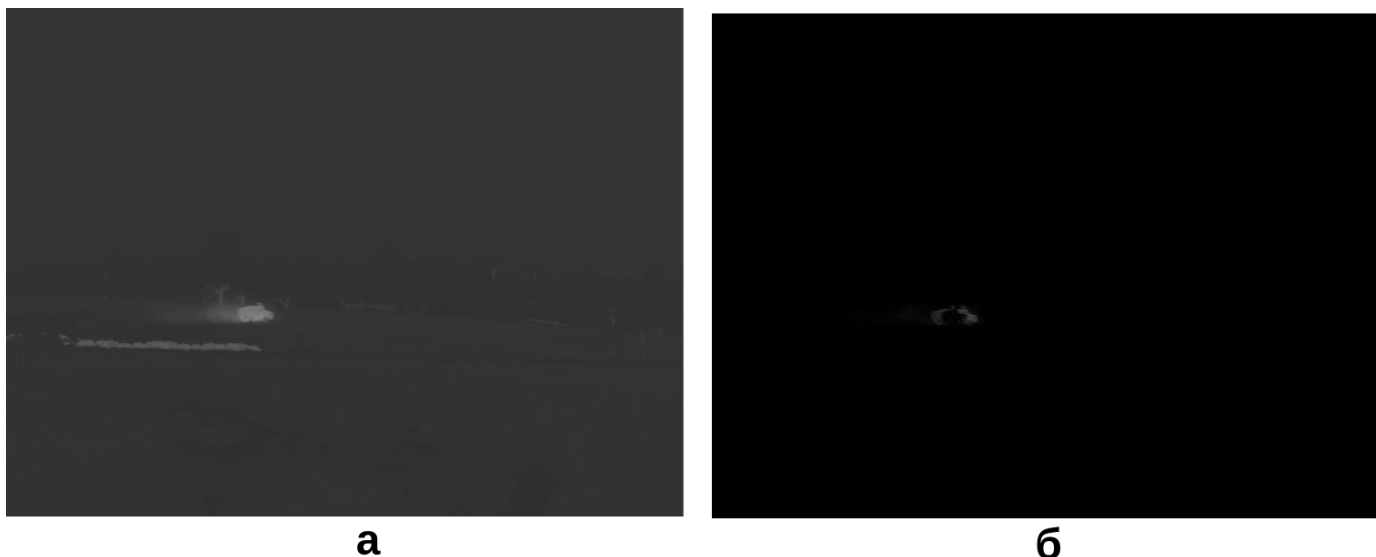


Рисунок 2.17 – Условно-реальные (синтетические) изображения различных типов: а) ИК-изображения; б) изображение с признаками движения

ненных методов комплексирования изображений можно отнести: поканальное слияние на уровне пикселей, метод главных компонент, дискретное вейвлет-преобразование и др.

Формирование разметки осуществляется на основе изображений с масками объектов, где интенсивности масок имеют уникальные значения, определяющие конкретные классы объектов. Такой подход позволяет разделить изображения объектов при их перекрытии относительно точки наблюдения. Координаты обрамляющих прямоугольников изображений объектов определяются по максимальным и минимальным значениям масок объектов x_{min} , y_{min} , x_{max} , y_{max} (рисунок 2.18). Каждое изображение объекта проходит проверку на условие малоразмерности. Если объект не удовлетворяет условиям малоразмерности, то изображение пропускается.

Конечным результатом является сохранение комплексированных изображений с текстовыми файлами аннотаций. В аннотациях содержатся уникальные номера объектов, индексы классов объектов и координаты обрамляющих прямоугольников.



Рисунок 2.18 – Отображение прямоугольника обрамляющего изображение маски объекта, и само изображение объекта интереса: а) исходное изображение; б) изображение с маской объекта

2.8 Формирование комплексированных изображений

Согласно разработанному алгоритму формирования аннотированных данных комплексированных изображений с признаками движения, необходимо в ручном режиме смоделировать 3D-карту местности, что требует от разработчика навыков владения Unreal Engine 5 и программирования на языке $C++$, с помощью которого происходит процесс взаимодействия со средой разработки. Формирование больших объёмов переменных данных возможно благодаря следующим ключевым особенностям:

1. Использование разнообразных сценариев и моделей.
2. Использование различных погодных и временных условий.
3. Широкий спектр материалов, разновидностей местности, текстур объектов.
4. Возможность изменения свойств объектов.
5. Автоматизация формирования аннотаций данных и др.

При формировании изображения видимого оптического диапазона используются вспомогательные каналы: карта глубины, карта нормалей, карта масок. На основе полученных характеристик сюжета можно синтезировать последовательность изображений новых типов: ИК-изображения, радиолокационные

изображения, видеокадры с признаками движения и др. В данной работе для повышения информативности системы признаков применяется предварительная обработка изображений, которая формирует псевдоцветное изображение с различительными признаками движения. Формирование комплексированных изображений с признаками движения осуществляется автоматически на основе программных компонент, реализующих разработанный алгоритм расширения признакового пространства изображений новым признаком движения (приложение A.1). Комплексирование разномодальных изображений осуществляется методом поканального слияния на уровне пикселей согласно разработанному алгоритму предварительной обработки изображений. Последующие шаги алгоритма формирования датасетов изображений с расширенным признаковым пространством выполняются в автоматическом режиме с помощью программных компонент, реализованных на языке программирования *Python*, приведенных в приложении A.2. Изображения объектов проверяются на условие малоразмерности. Выходные аннотированные изображения с признаками движения сохраняются в указанную директорию. Имена файлов аннотаций совпадают с именем изображений. Текстовый файл содержит информацию о классе изображений объекта и координаты обрамляющего прямоугольника:

$$object_{class}, \quad center_x, \quad center_y, \quad box_x, \quad box_y, \quad (2.6)$$

где $object_{class}$ – целочисленное значение, определяющее индекс класса объекта, начиная с 0, $center_x$ и $center_y$ – нормированные значения координат центра обрамляющего прямоугольника, box_x и box_y – нормированные значения ширины и высоты обрамляющего прямоугольника.

После формирования необходимого объёма данных условно-реальных комплексированных изображений с признаками движения необходимо разбить его на тренировочную и тестовую выборки таким образом, чтобы изображения в тренировочной и тестовой выборках были сформированы из разных сценариев, чтобы исключить предвзятость в оценке качества распознавания алгоритмов.

2.9 Выводы по материалам второй главы

В данной главе рассмотрено решение проблем, связанных с распознаванием малоразмерных изображений объектов. В ходе решения первой проблемы предложен метод расширения признакового пространства с целью повышения информативности системы признаков. Парадигма метода заключается в минимизации зон неоднозначности образов в признаковом пространстве путём увеличения расстояния между границами образов за счёт добавления различительных признаков движения. Чем больше расстояние между образами в признаковом пространстве, тем меньше количество ошибок классификации. Метод отличается от аналогов тем, что формирует изображение, в котором учтено больше различительных признаков движения, сформированных в процессе продолжительного наблюдения за объектом.

Для решения второй проблемы разработан алгоритм предварительной обработки изображений с целью комплексирования признаков движения. Алгоритм формирует изображение, в котором учтены исходные признаки изображения и новые признаки движения объектов. Алгоритм отличается тем, что формирует два разностных изображения, характеризующих наличие быстрого и медленного движения.

Проблема высокой стоимости разработки распознающей системы связана с высокими затратами процедуры формирования датасетов разномодальных изображений. Для решения данной проблемы разработан алгоритм формирования условно-реальных разномодальных изображений. Алгоритм позволяет в автоматическом режиме формировать датасеты, что сокращает стоимость процедуры формирования данных за счёт уменьшения времени на формирование данных, отсутствия необходимости наличия дорогостоящего записывающего оборудования и др.

Глава 3. РАЗРАБОТКА АЛГОРИТМОВ ПОСТРОЕНИЯ СВЁРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ НА УСТРОЙСТВАХ С ОГРАНИЧЕННЫМ ВЫЧИСЛИТЕЛЬНЫМ РЕСУРСОМ

3.1 Разработка структурной схемы алгоритма распознавания изображений с применением алгоритма предварительной обработки изображений на нейронном ускорителе

Концепция реализации алгоритмов СНС на устройствах с ограниченным вычислительным ресурсом заключается в переносе вычислительно сложных этапов алгоритма с центрального процессора на нейронный ускоритель с целью сокращения временной задержки выполнения алгоритма. К числу вычислительно сложных этапов алгоритмов распознавания малоразмерных изображений СНС является прямое распространение входного сигнала через экстрактор и агрегатор признаков, так как эти этапы алгоритма содержат большое количество вычислительно сложных операций (двумерная свёртка). Обычно предварительная обработка, декодирование признаков и постобработка не являются затратными этапами алгоритма, которые значительно снижают скорость обработки данных. В случаях, когда эти этапы алгоритма содержат вычислительно сложные операции, скорость обработки изображений высокого разрешения может существенно снизиться.

В решении практических задач предварительная обработка изображений высокого разрешения встречается довольно часто. Для обеспечения высоких показателей качества распознавания изображений необходимо использовать исходные изображения (без искажений) высокого разрешения, чтобы сохранить признаки малоразмерных изображений.

Решение четвертой проблемы. Алгоритм предварительной обработки изображений с целью комплексирования признаков движения выполняется на центральном процессоре, структурная схема которого представлена на рисун-

ке 1.11. Чтобы повысить скорость обработки данных алгоритма, предложена его реализация на нейронном ускорителе. Это достигается за счет объединения вычислений предварительной обработки изображений с экстрактором признаков СНС и переносом на вычислительные мощности нейронного ускорителя. С учётом модификации алгоритма предварительной обработки изображений структурная схема алгоритма распознавания изображений примет новый вид, представленный на рисунке 3.1.

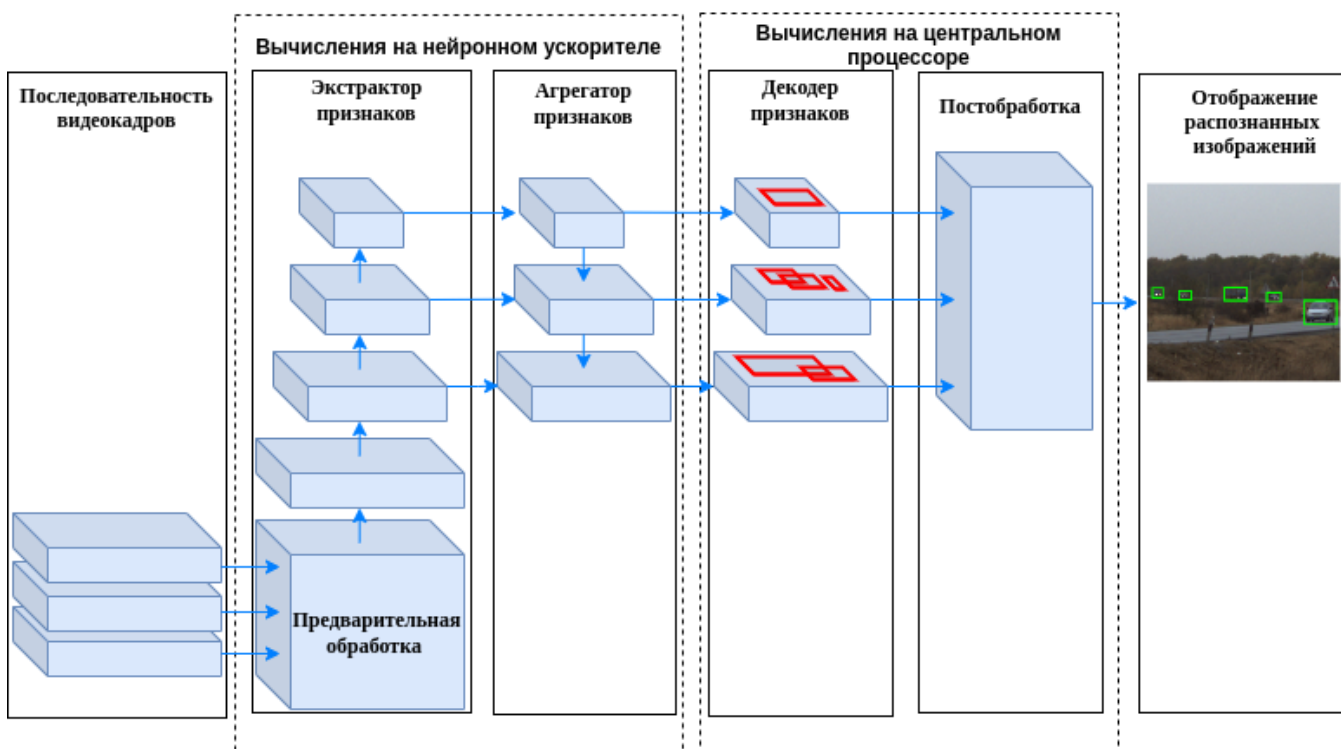


Рисунок 3.1 – Структурная схема алгоритма распознавания малоразмерных изображений объектов с учётом применения алгоритма предварительной обработки изображений на нейронном ускорителе

Далее рассмотрена процедура переноса алгоритма предварительной обработки изображений с центрального процессора на нейронный ускоритель.

3.2 Разработка алгоритма предварительной обработки изображений с целью комплексирования признаков движения на нейронном ускорителе

Согласно структурной схеме, представленной на рисунке 3.1, необходимо объединить последовательность выполнения операций, которые включает алгоритм предварительной обработки изображений с последовательностью операций, которые включает экстрактор признаков СНС. Для реализации данного подхода необходимо представить вычисления алгоритма предварительной обработки изображений в рамках операций, поддерживаемых нейронными ускорителями. К числу таких операций относятся: поэлементная (попиксельная) матричная разность и вычисление абсолютных величин.

На рисунке 3.2 представлена структурная схема алгоритма предварительной обработки изображений с целью комплексирования признаков движения в рамках операций, поддерживаемых нейронными ускорителями. Алгоритм формирует комплексированное изображение с двумя разностными каналами на основе опорного изображения и двух изображений, задержанных на T_0 и $4T_0$.

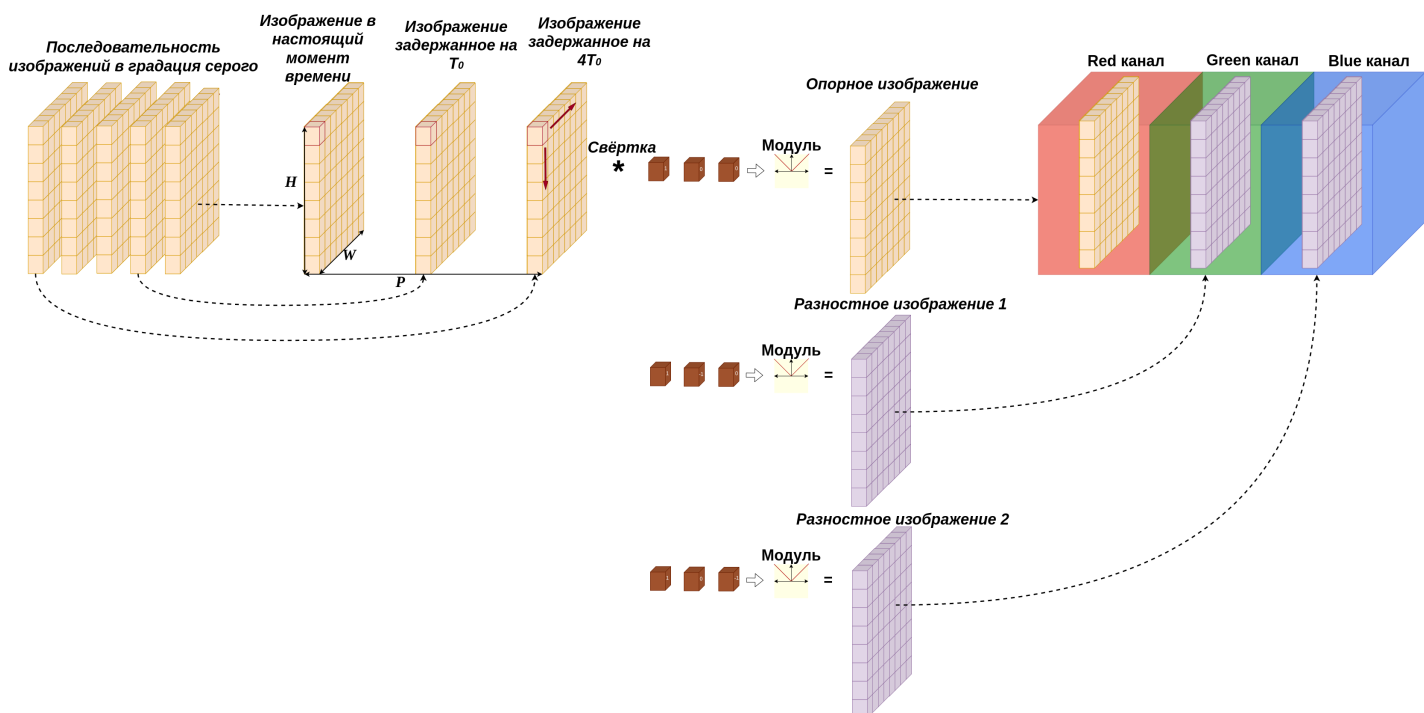


Рисунок 3.2 – Структурная схема алгоритма предварительной обработки изображений с целью комплексирования признаков движения на нейронном ускорителе

С учётом выбранных задержек для получения комплексированного изображения необходимо накопить последовательность из пяти изображений. На основе накопленной последовательности формируется тензор из изображений, полученных в настоящий момент времени и задержанных на T_0 и $4T_0$. Формирование опорного изображения и двух разностных изображений реализовано с помощью операции двумерная свёртка с размерностью фильтров $3 \times 3 \times 1 \times 1$ и значений фильтров, состоящих из 0, 1, -1.

Единичные значения фильтров формируют опорное изображение, отрицательные значения формируют задержанное изображение с отрицательными значениями, нулевые значения удаляют все признаки изображения. Очередность значений фильтра 1, 0, 0 формирует опорное изображение. Очередность значений фильтра 1, -1, 0 формирует разностное изображение, характеризующее наличие медленного движения, а фильтр со значениями 1, 0, -1 формирует разностное изображение, характеризующее наличие быстрого движения. Выполнение алгоритма можно представить в виде матричных преобразований (свёртка):

$$Y = \sigma(X * W) = |X * W|, \quad (3.1)$$

где Y – трёхмерная матрица выходных значений; X – трёхмерная матрица входных значений; σ – функция активации абсолютных величин; W – трёхмерная матрица весовых коэффициентов. Таким образом, на выходе свёртки создаётся тензор, равный по размерности входному тензору.

Опорное изображение имеет только положительные значения, а значения изображений в разностных каналах могут быть отрицательными. Чтобы привести значения выходного тензора к одному динамическому диапазону от 0 до 1 во всех каналах комплексированного изображения, определяются абсолютные величины. Результатом всех операций является тензор, состоящий из опорного и двух разностных изображений. Если представить комплексированное изображение в цветовом пространстве RGB, то опорное изображение находится в цветовом канале Red, а разностные изображения – в цветовых каналах Green и Blue.

На основе структурной схемы алгоритма предварительной обработки изображений реализована *Pytorch* модель алгоритма на языке программирования *Python*. Листинг программных компонент приведен в приложении А.4. Программная реализация алгоритма имеет возможность экспортировать *Pytorch* модель алгоритма предварительной обработки изображений в унифицированный формат Open Neural Network Exchange (ONNX), который поддерживается большинством фреймворков глубокого обучения.

Рассмотрим основные части программной реализации алгоритма предварительной обработки изображений. Входные изображения приводятся к типу данных с плавающей точкой *float32* и нормализуются. Операция двумерная свёртка реализуется с помощью функции

$$\text{torch.nn.Conv2d}(in_channels, out_channels, kernel_size), \quad (3.2)$$

где *in_channels* – входное количество каналов, *out_channels* – выходное количество каналов, *kernel_size* – размер фильтров. Вычисление абсолютных величин тензора реализуется с помощью функции

$$\text{torch.abs}(input), \quad (3.3)$$

где *input* – входной тензор.

3.3 Разработка алгоритма построения системы распознавания малоразмерных изображений на устройствах с ограниченным вычислительным ресурсом

Последней тенденцией развития вычислительных устройств является аппаратное оснащение нейронным ускорителем. Современные нейронные ускорители поддерживают вычисления с различной разрядностью. Быстродействие алгоритма напрямую зависит от разрядности вычислений. Среди широко распространенных разрядностей вычислений можно выделить Floating Point 32 (FP32), Floating

Point 16 (FP16), Integer 8 (INT8). Переход к вычислениям с пониженной разрядностью приводит к увеличению скорости выполнения алгоритма, но «платой» за это является снижение качества распознавания изображений объектов [127, 128]. В определенных случаях можно «пожертвовать» качеством распознавания в угоду увеличения скорости обработки данных за счёт перехода к вычислениям с пониженной разрядностью.

Вычислительные устройства могут поддерживать несколько режимов энергопотребления. Каждый режим энергопотребления задействует определенные ресурсы вычислительного устройства. Обычно, режим максимального энергопотребления задействует все ресурсы вычислительного устройства, обеспечивая наибольшую производительность.

Как правило, распознающие системы располагаются на автономных носителях, у которых ограничен энергетический потенциал, что накладывает ограничения на предоставляемую потребляемую мощность. Поэтому, одной из важных особенностей при выборе вычислительного устройства является потребляемая мощность. С одной стороны, потребляемая мощность устройства напрямую влияет на производительность устройства, что ограничивает применение вычислительно сложных архитектур СНС. С другой стороны, обеспечение высоких показателей качества распознавания достигается вычислительно сложными архитектурами СНС. Поэтому скорость обработки данных, качество распознавания и потребляемая мощность устройства являются взаимозависимыми характеристиками. Для достижения необходимых показателей распознающей системы необходимо определить набор компонент и параметров вычислительного устройства для построения программного комплекса, который позволит достигнуть компромисса между требуемыми характеристиками.

Решение пятой проблемы. С учётом сформулированных замечаний разработан алгоритм формирования компонент, на основе которых можно построить аппаратно-программный комплекс или далее распознающую систему, удовлетворяющую необходимым характеристикам (рисунок 3.3). Комментарии к структурной схеме алгоритма [129]:

1. *Формирование исходного состава компонент и требуемых характеристик распознающей системы.*

а) *Требуемые характеристики.* На данном этапе задаются необходимые характеристики распознающей системы: время выполнения алгоритма, качество распознавания малоразмерных изображений по заданному критерию, допустимая потребляемая мощность вычислительного устройства. Требуемые характеристики определяются решаемыми задачами. СНС выполняет функции компонентов системы распознавания, поэтому требования к СНС соответствуют требованиям к системе в целом. Обычно время на распознавание изображения объекта не должно превышать 40 мс (масштаб реального времени 25 кадр/с). Предел предоставляемой потребляемой мощности вычислительному устройству индивидуален и зависит от энергетического запаса автономного носителя. Качество распознавания определяется исключительно из целевой задачи и специфики применения системы.

б) *Формирование множества вычислительных устройств.* Множество вычислительных устройств может быть сформировано на основе различных требований. К числу таких требований можно отнести допустимую потребляемую мощность вычислительного устройства, наличие аппаратной и программной поддержки нейронного ускорителя и др.

в) *Формирование множества архитектур СНС.* Чтобы обеспечить необходимое качество распознавания и время выполнения алгоритма целесообразно использовать одноэтапные архитектуры СНС. Операции, входящие в состав архитектуры СНС, вычислительная сложность архитектуры, качество распознавания на открытых датасетах являются важными факторами, влияющими на формирование множества архитектуры СНС. Определённые операции, входящие в состав СНС могут затрачивать неприемлемый объём вычислительного ресурса или быть вовсе нереализуемы на выбранном вычислительном устройстве, что является основанием игнорировать архитектуру.

г) *Формирование данных.* Для обучения и оценки качества распознавания СНС необходимо сформировать тренировочный и тестовый датасеты. Для калибровки моделей СНС используется калибровочный датасет изображений.

2. *Формирование набора.* Из множества вычислительных устройств выбирается непроанализированное вычислительное устройство с программной поддержкой. Если рассмотрены все вычислительные устройства, то происходит переход к шагу 3.

а) *Выбор архитектуры СНС.* Из множества архитектур СНС выбирается непроанализированная архитектура СНС. Если рассмотрены все архитектуры СНС, то происходит переход к шагу 2.

б) *Обучение модели СНС.* На базе тренировочного датасета происходит обучение СНС. Обучение проводится на вычислительных машинах, содержащих массивы графических процессоров. Основанием выбора модели СНС являются показатели качества распознавания.

в) *Выбор разрядности вычислений.* На этом этапе алгоритма происходит выбор разрядности вычислений, который поддерживается нейронным ускорителем. Если рассмотрены все поддерживаемые разрядности вычислений для данного устройства, то происходит переход к шагу 2 а.

г) *Построение модели СНС.* На выбранном вычислительном устройстве происходит построение обученной модели СНС с учётом выбранной разрядности вычислений. Для повышения скорости обработки данных СНС применяется переход к вычислениям с пониженной разрядностью. Вычисление в целочисленных значениях INT8 требует калибровки модели. В результате калибровки происходит квантование весовых коэффициентов модели СНС и признаков на выходе активаций. Квантование можно осуществить двумя способами Post-training Quantization (PTQ) и Quantization-aware Training (QAT). PTQ – это упрощенный метод квантования, который выполняется после обучения. Поскольку есть доступ к весовым коэффициентам модели СНС, можно провести их квантование по плотности распределения. Квантование активаций является более сложной задачей, поскольку распределения признаков на выходе активаций необходимо измерять

с использованием реальных данных. Для этого в процессе прямого распространения реальных данных собирается статистика о межуровневых распределениях признаков на выходе активаций обученной модели НС и определяются коэффициенты масштабирования. Этот процесс представляет собой калибровку, а используемый репрезентативный набор данных называется калибровочным датасетом.

Если РТQ приводит к значительному снижению качества распознавания, то применяется метод квантования QAT. QAT проводит квантование на этапе обучения, что позволяет избежать значительного снижения качества распознавания изображений объектов. Квантование методом QAT необходимо начинать с предварительно обученной модели НС.

Квантование РТQ является более популярным методом, поскольку он более прост и быстр в применении. QAT почти всегда обеспечивает лучшее качество распознавания СНС, а иногда это единственный способ для достижения необходимого качества распознавания.

Если построить модель на данном вычислительном устройстве не представляется возможным, то происходит переход к шагу 2 а. Ограниченный объем необходимой памяти, отсутствие программной поддержки операций, входящих в состав архитектуры СНС и другие причины, могут являться ограничением при построении модели СНС.

д) *Оценка времени выполнения СНС.* На этом шаге происходит оценка времени выполнения модели СНС на вычислительном устройстве. В случае неудовлетворительного времени выполнения модели происходит переход к шагу 2 в.

е) *Оценка качества распознавания изображений объектов.* Оценка качества распознавания изображений объектов проводится на тестовом датасете с помощью заданного критерия качества распознавания. Если качество распознавания модели СНС удовлетворяет необходимым характеристикам, то формируется набор из вычислительного устройства с программной поддержкой, архитектуры СНС, датасетов и параметра нейронного ускорителя: разрядность вычислений. В

случае неудовлетворительного качества распознавания модели СНС происходит переход к шагу 2 а.

ж) *Буферизация набора*. Если все характеристики пространной системы удовлетворяют заданным критериям, то набор буферизируется.

3. *Выбор набора*. После того как будут рассмотрены все удовлетворяющие комбинации, наступает заключительный этап выбора лучшего набора на основе полученных оценок характеристик распознающей системы методом взвешенной суммы нормированных неоднородных критериев [130]:

$$F(\hat{f}|w) = w_0\hat{f}_0 + \dots + w_m\hat{f}_m, \quad (3.4)$$

где w – весовые коэффициенты (коэффициенты важности) критерия, \hat{f} – нормированные значения неоднородных критериев. Каждому критерию необходимо присвоить свой коэффициент важности. В сумме все коэффициенты важности дают единицу. Поскольку критерии имеют разные шкалы в силу своей природы, то значения критериев необходимо привести к сопоставимому виду, нормализуя их. Нормализация критериев f_i осуществляется на основе линейной функции:

$$\hat{f}_i = \frac{f_i}{f_{i*}}, \quad (3.5)$$

где f_{i*} – наибольшее значение критерия. Таким образом, значения нормированных критериев находятся в диапазоне от 0 до 1. Поскольку каждое большее значение критерия должно быть предпочтительнее, то возьмём обратные значения для оценок времени выполнения и номинальных значений потребляемой мощности вычислительных устройств. На основе формулы (3.4) формируются обобщенные показатели значимости F для всех наборов. На основе рассчитанных показателей значимости выбирается набор с наибольшим значением.

С помощью сформированного набора можно построить интеллектуальную систему распознавания малоразмерных изображений, удовлетворяющую заданным характеристикам. На рисунке 3.4 приведена структурная схема алгоритма

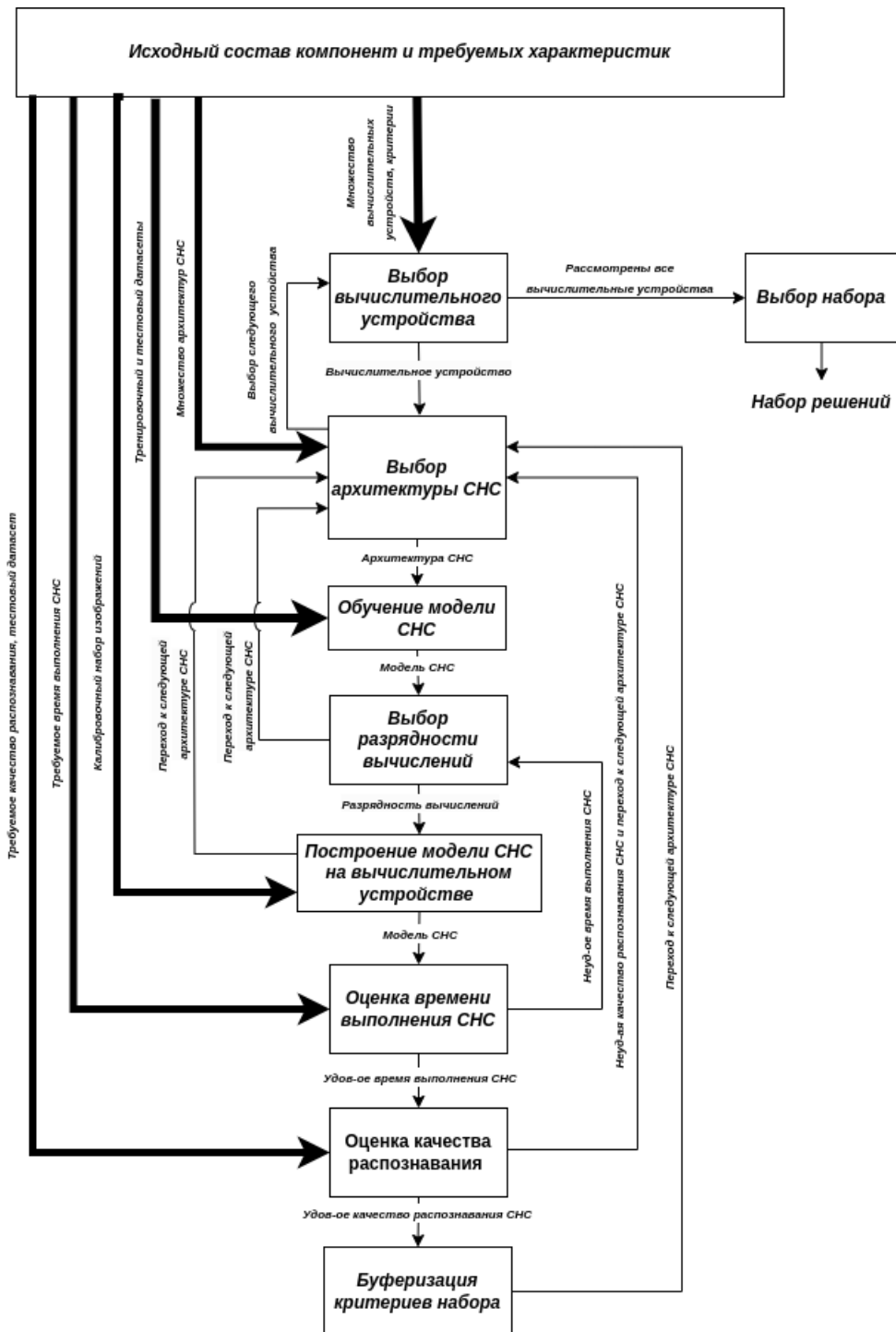


Рисунок 3.3 – Структурная схема алгоритма формирования набора для создания распознающей системы

построения интеллектуальной системы. Далее приведены комментарии к структурной схеме алгоритма.

1. Набор для создания распознающей системы состоит: 1) из вычислительного устройства с программой поддержкой; 2) архитектуры СНС; 3) тренировочного, тестового и калибровочного датасетов; 4) разрядности вычислений.

2. *Обучение модели СНС.* На основе архитектуры СНС строится модель, далее проводится обучение модели на тренировочном датасете. Выбирается модель, обеспечивающая лучший показатель качества распознавания на тестовом датасете.

3. *Переход к вычислениям с выбранной разрядностью.* На этом этапе проводится формирование модели СНС с учётом выбранной разрядности вычислений. Для целочисленной разрядности вычислений проводится калибровка модели с помощью калибровочного датасета и программного обеспечения.

4. *Построение модели СНС на вычислительном устройстве.* Завершающим этапом является построение сформированной модели СНС на вычислительном устройстве с помощью программного обеспечения.

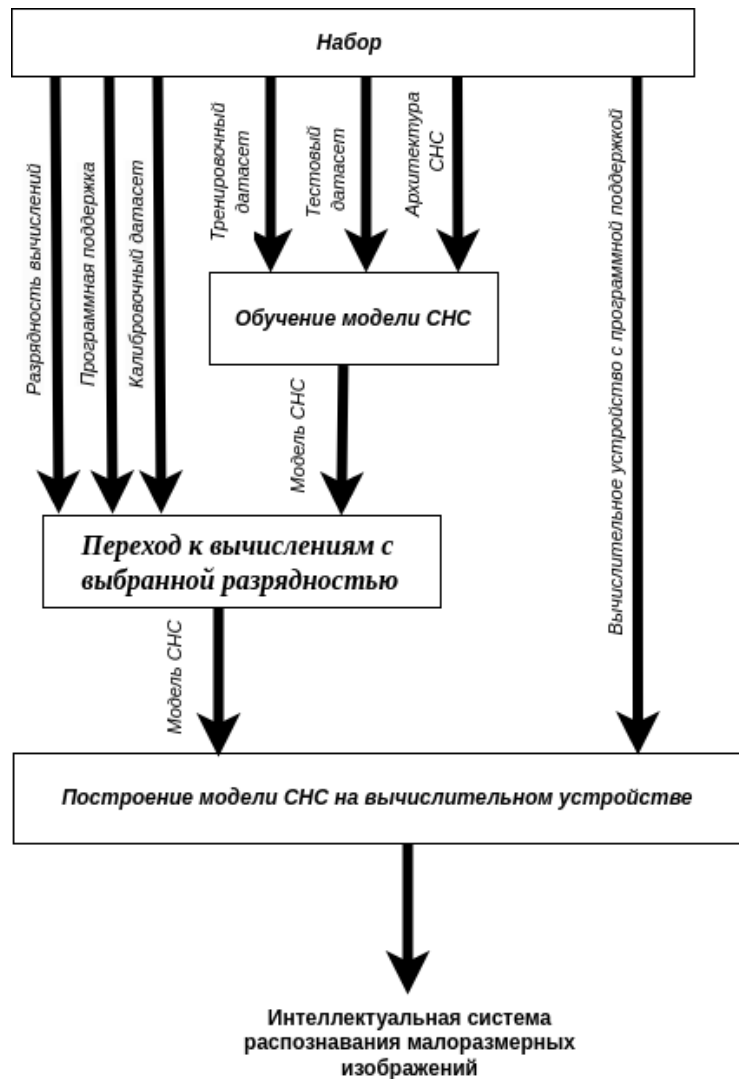


Рисунок 3.4 – Алгоритм построения интеллектуальной системы распознавания малоразмерных изображений объектов

3.4 Выводы по материалам третьей главы

В данной главе рассмотрено решение четвертой и пятой проблем, связанных с построением и выполнением свёрточных нейронных сетей на устройствах с ограниченным вычислительным ресурсом.

В ходе решения четвертой проблемы, связанной с неудовлетворительной временной задержкой выполнения алгоритма предварительной обработки изображений с целью комплексирования признаков движения на устройствах с ограниченным вычислительным ресурсом, предложен алгоритм, который переносит выполнение алгоритма на вычислительные мощности нейронного ускорителя. Это достигается за счёт присоединения алгоритма предварительной обработки изображений к экстрактору признаков свёрточной нейронной сети с последующим выполнением на нейронном ускорителе.

Для решения проблемы, связанной с построением интеллектуальной распознающей системы, удовлетворяющей заданным характеристикам, разработан алгоритм, который формирует набор, состоящий: из вычислительного устройства с программной поддержкой, архитектуры свёрточной нейронной сети, разрядности вычислений нейронного ускорителя, датасетов. С помощью определенной последовательности действий над составными частями набора можно построить распознающую систему, обеспечивающую необходимые характеристики в качестве распознавания малоразмерных изображений объектов, скорости обработки данных и потребляемой мощности устройства, не превышающей допустимые значения.

Глава 4. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ РАЗРАБОТАННЫХ МЕТОДА И АЛГОРИТМОВ

4.1 Вычислительный эксперимент с целью исследования метода расширения признакового пространства и алгоритма предварительной обработки изображений

4.1.1 Описание вычислительного эксперимента с целью исследования метода расширения признакового пространства и алгоритма предварительной обработки изображений

В этой главе рассматривается экспериментальное исследование влияния разработанного метода расширения признакового пространства на информативность системы признаков и алгоритма предварительной обработки изображений на качество распознавания малоразмерных изображений одноэтапными архитектурами СНС с помощью проведения вычислительного эксперимента. В качестве одноэтапных архитектур СНС взят тестовый набор перспективных СНС с различной вычислительной сложностью (таблица 3).

Таблица 3 – Тестовые архитектуры свёрточных нейронных сетей

Архитектура	Экстрактор признаков	Агрегатор признаков	Декодер признаков
YOLOv3-Tiny	Darknet19	FPN	YOLO
YOLOv5s	YOLOv5s	PAN	YOLO
YOLOv5l	YOLOv5l	PAN	YOLO
YOLOv8s	YOLOv8s	PAN	YOLO

Для исследования разработанных метода и алгоритма сформированы три датасета аннотированных условно-реальных изображений с помощью разработанного алгоритма формирования условно-реальных изображений. Первый датасет состоит из изображений в градациях серого, второй датасет из комплексированных изображений с одним разностным каналом, а третий датасет с двумя

разностными каналами. Каждый датасет был разбит на тренировочную и тестовую выборки.

Все тестовые архитектуры СНС обучены на трех тренировочных выборках. После чего проведена оценка информативности системы признаков на основе величины потерь ошибок распознавания обученных СНС на тестовом датасете. В качестве критерия величины потерь используется оценка частоты ошибок распознавания *ErrorRate*. Также проведена оценка качества распознавания обученных СНС на тестовом датасете с помощью критерия *mAP* и вероятностных характеристик распознавания.

4.1.2 Формирование датасета для задачи распознавания малоразмерных изображений объектов

На основе разработанного алгоритма формирования аннотированных данных условно-реальных изображений получено 80 аннотированных последовательностей синтетических видеокадров видимого оптического диапазона. Каждая последовательность видеокадров имеет индивидуальный статичный фон с присутствием малоразмерных изображений подвижных и неподвижных объектов. Датасет содержит объекты 6 классов автомобильной техники. Видеокадры представлены в градациях серого от 0 до 255 и имеют разрешение 960 на 960 элементов с частотой появления 25 кадр/с. Минимальный размер изображений целевых объектов находится в пределах 8-14 пикселей. На рисунке 4.1 представлены примеры изображений из сформированных последовательностей видеокадров.

На основе исходных последовательностей видеокадров сформированы три датасета аннотированных изображений: изображения в градациях серого, комплексированные изображения с одним разностным каналом и двумя разностными каналами.

Вычислительная сложность модели СНС зависит от размерности входного изображения. Чтобы соблюсти одинаковую вычислительную сложность тестовых архитектур необходимо установить одинаковую размерность входного

**а****б**

Рисунок 4.1 – Примеры изображений из сформированных последовательностей видеок кадров

тензора. Поэтому в вычислительном эксперименте каждое изображение представлено в цветовом пространстве RGB. На рисунке 4.2 приведена иллюстрация формирования изображения для датасетов: а) изображений в градациях серого; б) комплексированных изображений с одним разностным каналом (каналы Red и Green содержат опорные изображения, канал Blue содержит разностное изображение); в) комплексированных изображений с двумя разностными каналами (канал Red содержит опорное изображение, каналы Green и Blue содержат разностные изображения).

Сформированные датасеты аннотированных изображений разбиты на тренировочные и тестовые выборки. Для достижения релевантности экспериментальных результатов датасеты сформированы таким образом, чтобы изображения с одной видеопоследовательности не входили в обе выборки. На основе первых 50 видеопоследовательностей сформированы тренировочные выборки 150000 изображений, а на остальных 30 видеопоследовательностях тестовые выборки 90000 изображений.

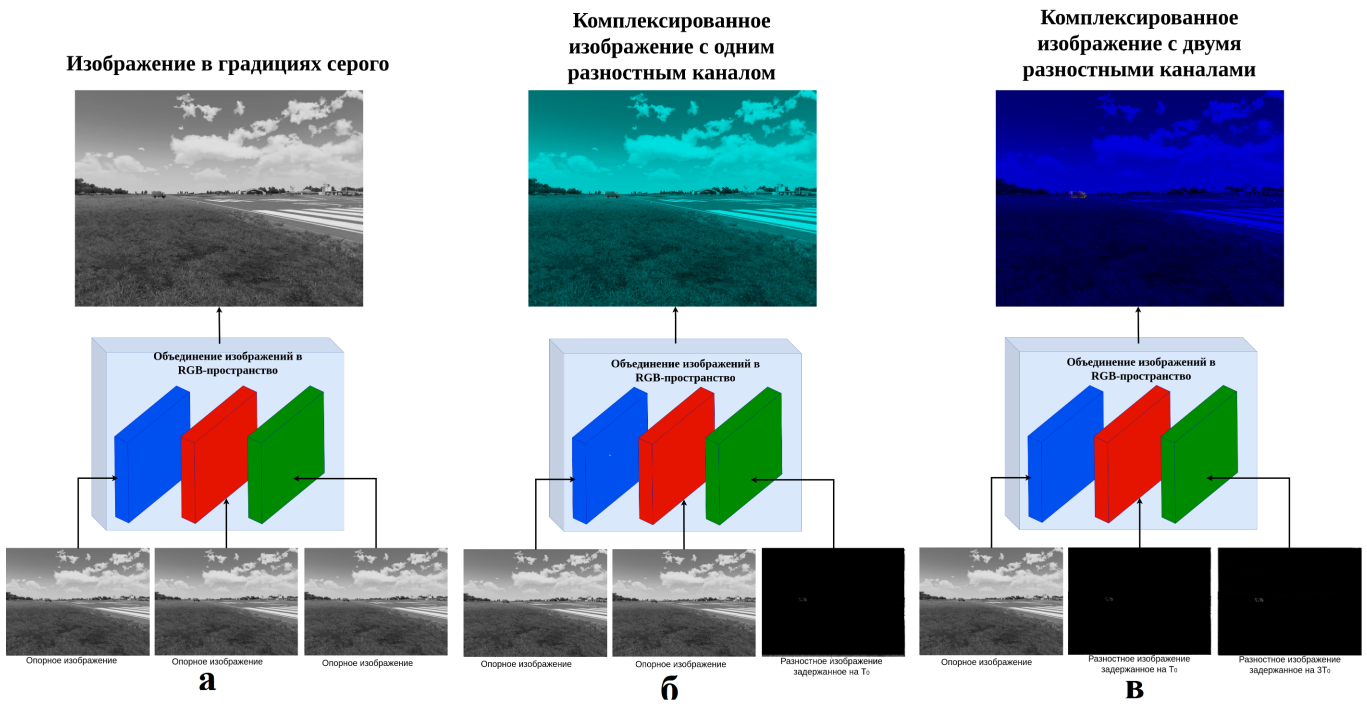


Рисунок 4.2 – Иллюстрация формирования датасетов изображений

4.1.3 Обучение тестовых архитектур свёрточных нейронных сетей

Обучение СНС осуществлялось на вычислительной машине, которая состоит из графического процессора NVIDIA GeForce RTX 2080 Ti, центрального процессора Intel Core i7-6900K CPU 3.20GHz, 64 Гб оперативной памяти RAM 64GiB DIMM Synchronous 2133 MHz.

Обучение всех тестовых моделей проводилось на основе фреймворка *Ultralytics* [114], реализованного на языке программирования *Python* с помощью библиотеки глубокого обучения *Pytorch*. Фреймворк *Ultralytics* включает широкий спектр инструментов от обучения, тонкой настройки моделей, оптимизации вычислений прямого вывода и до интеграции передовых архитектур СНС на встраиваемые вычислительные устройства. Для повышения обобщающей способности в процессе обучения используются различные подходы регуляризации: многомасштабное обучение, аугментация данных, передачи знаний (Transfer Learning) и другие подходы.

Перед обучением моделей СНС настроен конфигурационный файл архитектуры СНС, в котором необходимо указать количество классов интереса.

Вычисление опорных прямоугольников проводится автоматически на основе аннотаций тренировочного датасета методом k -средних [131]. Для выделения семантических представлений малоразмерных изображений объектов на высоких уровнях пирамиды признаков используются изображения высокого разрешения: ширина изображения 960, высота изображения 960, глубина 3. Размерность тензора на выходе агрегатора признаков определяется автоматически. Вычислительная сложность тестовых моделей СНС зависит от размера входных данных и измеряется в количестве операций с плавающей точкой в секунду или Floating-Point Operations per Second (FLOPS).

В большинстве случаев инициализация весовых коэффициентов модели СНС осуществляется случайным образом по нормальному закону распределения. В данном вычислительном эксперименте при обучении модели СНС используется метод передачи знаний (Transfer Learning). Суть Transfer Learning заключается в инициализации коэффициентов экстрактора признаков коэффициентами СНС, обученной на крупномасштабном классификационном датасете ImageNet, который включает 1200000 изображений 1000 классов. Такой подход позволяет сформировать у СНС визуальное представление об образах объектов реального мира, что повышает способность к обобщению СНС [132].

Далее каждая тестовая модель СНС была обучена на трех датасетах. Ниже на рисунках 4.3 – 4.5 приведены зависимости значений ошибок обучения тестовых моделей СНС от эпох обучения на трех датасетах. В таблицах 4 – 7 отражены основные характеристики обучения тестовых моделей СНС на трех датасетах.

Обучение моделей СНС YOLOv3-Tiny.

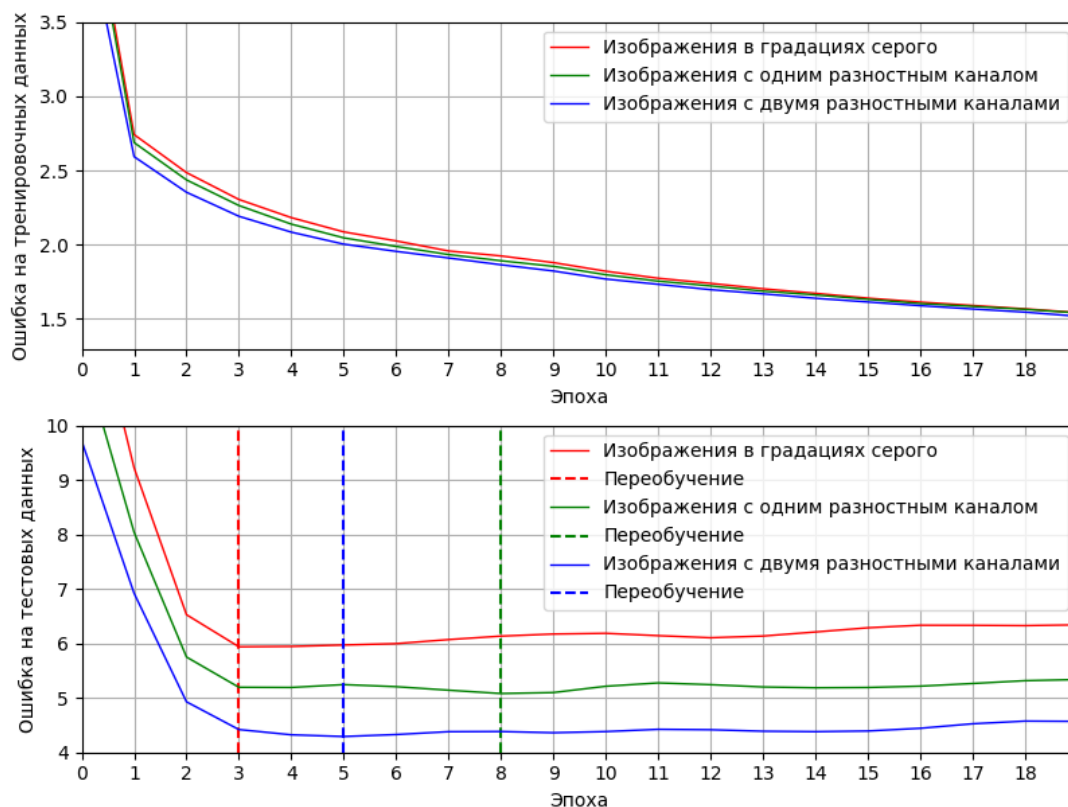


Рисунок 4.3 – Зависимости значений ошибки обучения моделей СНС YOLOv3-Tiny от эпох обучения

Таблица 4 – Характеристики обучения тестовых моделей СНС YOLOv3-Tiny

Характеристики	Изображения в градациях серого	Комплексированные изображения с одним разностным каналом	Комплексированные изображения с двумя разностными каналами
Размер данных	3 x 960 x 960		
Размер батча	32		
Вычислительная сложность, 10^9 FLOPs	19		
Время обучения одной эпохи, мин	14		
Время обучения, час	4.66		
Необходимая память GPU, Гбайт	10.6		
Минимальное значение ошибки обучения на тестовом датасете	5.9	5.1	4.4

Обучение моделей СНС YOLOv5s.

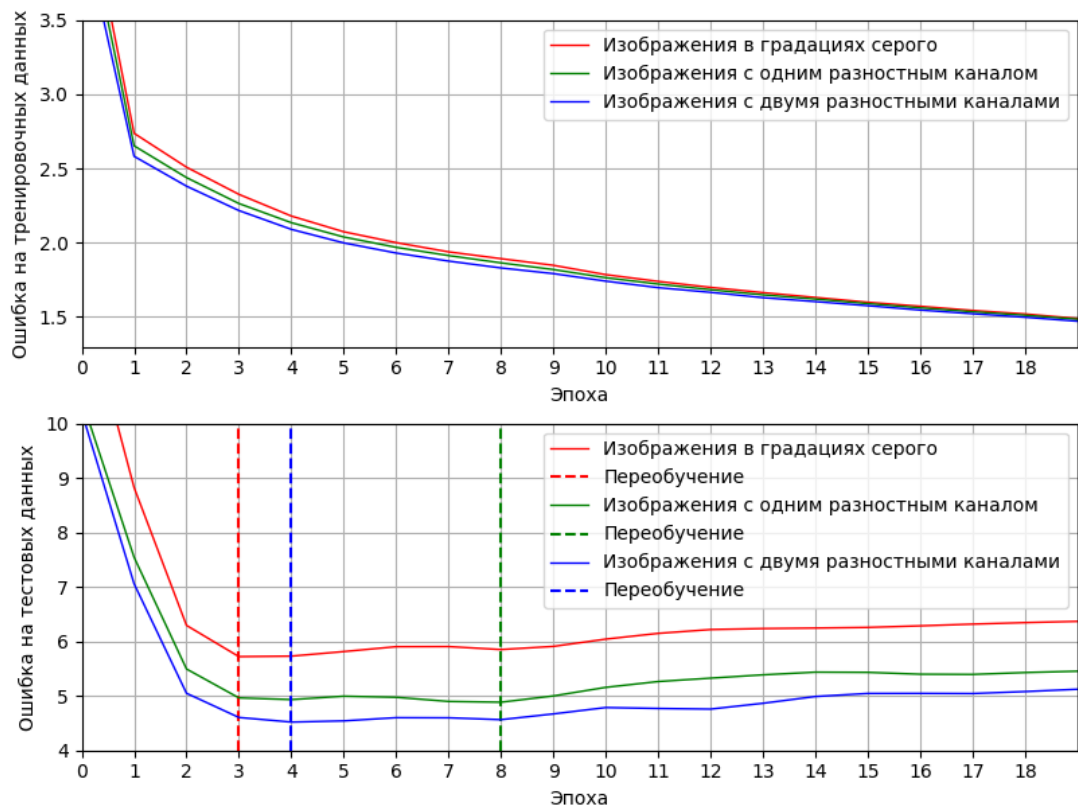


Рисунок 4.4 – Зависимости значений ошибки обучения моделей СНС YOLOv5s от эпох обучения

Таблица 5 – Характеристики обучения тестовых моделей СНС YOLOv5s

Характеристики	Изображения в градациях серого	Комплексированные изображения с одним разностным каналом	Комплексированные изображения с двумя разностными каналами
Размер данных	3 x 960 x 960		
Размер батча	16		
Вычислительная сложность, 10^9 FLOPs	24.1		
Время обучения одной эпохи, мин	21		
Время обучения, час	7		
Необходимая память GPU, Гбайт	8.6		
Минимальное значение ошибки обучения на тестовом датасете	5.8	4.9	4.6

Обучение моделей СНС YOLOv5l.

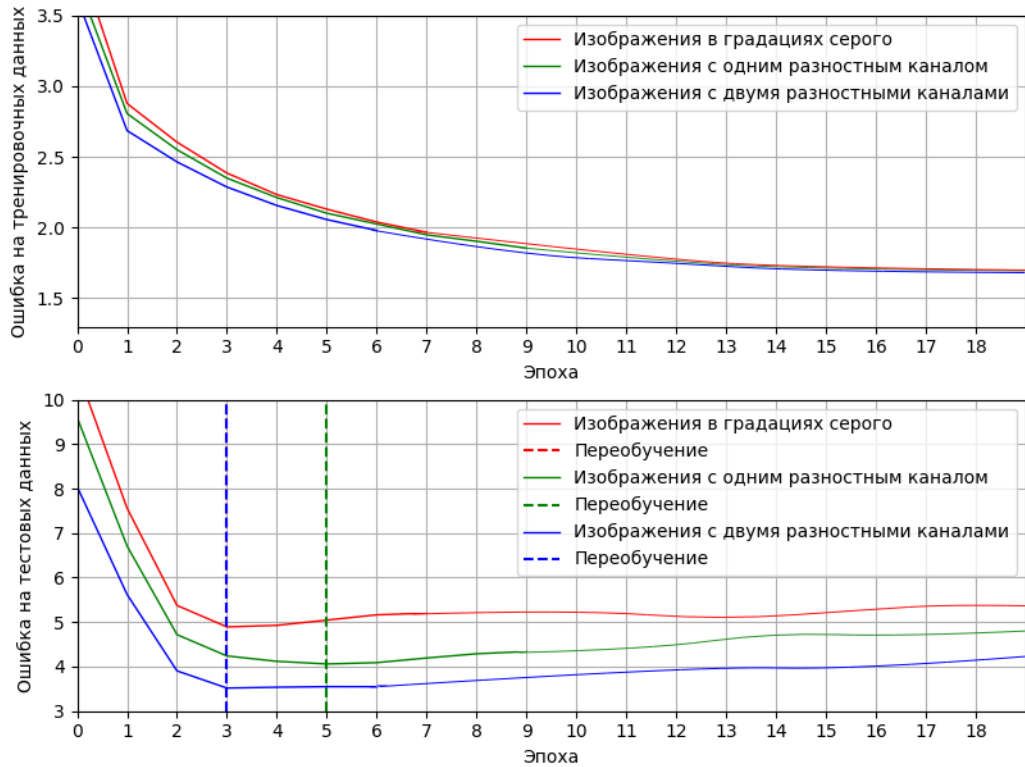


Рисунок 4.5 – Зависимости значений ошибки обучения моделей СНС YOLOv5l от эпох обучения

Таблица 6 – Характеристики обучения тестовых моделей СНС YOLOv5l

Характеристики	Изображения в градациях серого	Комплексированные изображения с одним разностным каналом	Комплексированные изображения с двумя разностными каналами
Размер данных	3 x 960 x 960		
Размер батча	8		
Вычислительная сложность, 10^9 FLOPs	135.3		
Время обучения одной эпохи, мин	65		
Время обучения, час	21.6		
Необходимая память GPU, Гбайт	11.3		
Минимальное значение ошибки обучения на тестовом датасете	4.9	4.3	3.6

Обучение моделей СНС YOLOv8s.

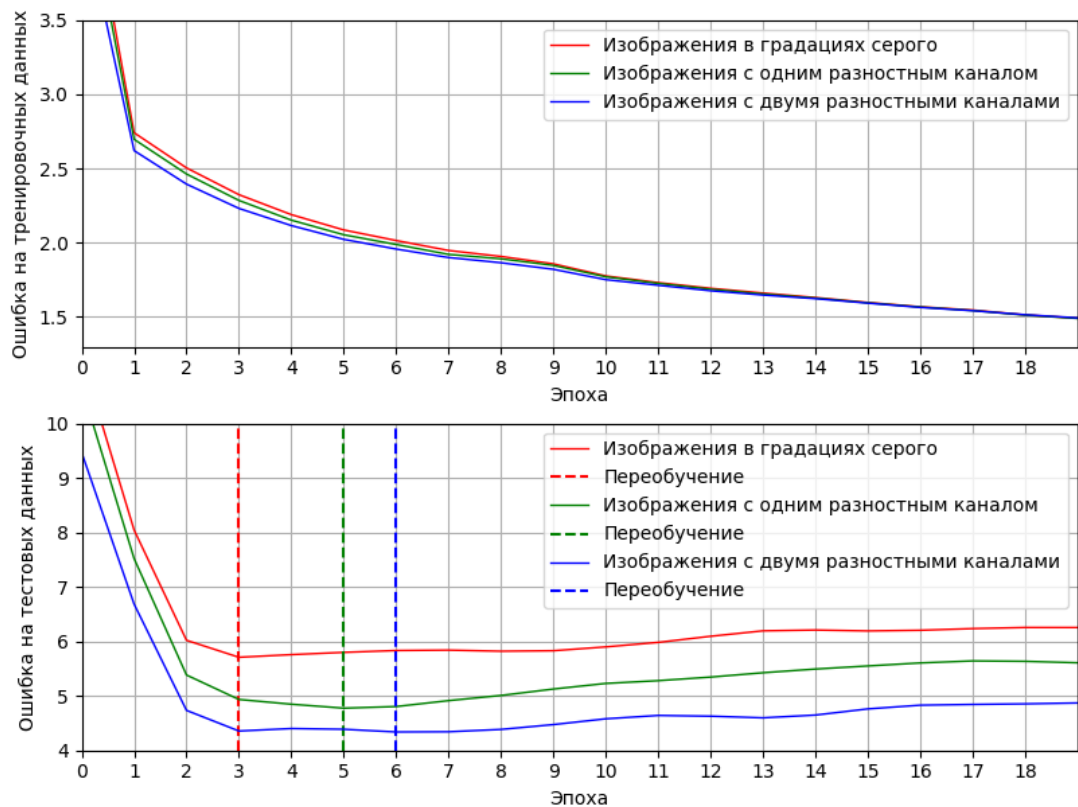


Рисунок 4.6 – Зависимости значений ошибок обучения архитектуры СНС YOLOv8s от эпох обучения

Таблица 7 – Характеристики обучения тестовых моделей СНС YOLOv8s

Характеристики	Изображения в градациях серого	Комплексированные изображения с одним разностным каналом	Комплексированные изображения с двумя разностными каналами
Размер данных	3 x 960 x 960		
Размер батча	16		
Вычислительная сложность, 10^9 FLOPs	28.7		
Время обучения одной эпохи, мин	22		
Время обучения, час	7.3		
Необходимая память GPU, Гбайт	7.98		
Минимальное значение ошибки обучения на тестовом датасете	5.7	4.8	4.3

4.1.4 Оценка информативности системы признаков

Для оценки информативности системы признаков, сформированных датасетов изображений, используется частота ошибок *ErrorRate* (формула (1.10)) обученных на этих датасетах СНС. Варьированием порога уверенности решения алгоритма формируется множество значений *ErrorRate*, из которого выбирается наименьшее значение. Оценки *ErrorRate* для тестовых архитектур представлены в таблице 8.

Таблица 8 – Оценки частоты ошибок распознавания *ErrorRate* СНС

Датасет	<i>ErrorRate</i> (YOLOv3-Tiny), %	<i>ErrorRate</i> (YOLOv5s), %	<i>ErrorRate</i> (YOLOv5l), %	<i>ErrorRate</i> (YOLOv8s), %
Изображения в градациях серого	47.6	39.6	23.1	37.5
Комплексированные изображения с одним разностным каналом	41.9	31.9	15.7	28.8
Комплексированные изображения с двумя разностными каналами	33.1	27.3	11.6	25.4

СНС, обученные на датасетах, с двумя разностными каналами достигли меньших значений частоты ошибок распознавания *ErrorRate* на 3.4 – 8.8 %, чем обученные на датасетах с одним разностным каналом и на 11.5 – 14.5 %, чем обученные на датасетах исходных изображений. Исходя из того, что информативность системы признаков отображает величина ошибок распознавания СНС или же частота ошибок распознавания, то признаковое пространство, представленное изображениями с двумя разностными каналами, имеет большую информативность. Таким образом, предложенный метод расширения исходного признакового пространства с помощью добавления признаков движения для нескольких моментов времени позволил сформировать систему признаков, обладающую большей информативностью.

4.1.5 Оценка качества распознавания малоразмерных изображений объектов свёрточных нейронных сетей

Для оценки качества распознавания обученных СНС используются критерии mAP и вероятности правильного распознавания при фиксированной частоте ложной тревоги $P_{\text{рас}}|_{F_{\text{лт}}=\text{const}}$. Оценка качества распознавания всех тестовых моделей СНС проводилась на изображениях одного и того же размера $960 \times 960 \times 3$.

На рисунках 4.7 – 4.10 приведены зависимости $Precision - Recall$ для обученных тестовых архитектур. На рисунке 4.11 приведены оценки интегральных характеристик качества распознавания mAP (площадь под кривой $Precision - Recall$) для всех тестовых моделей СНС. На рисунках 4.12 – 4.15 приведены зависимости оценки вероятности правильного распознавания от частоты ложной тревоги обученных моделей СНС. На рисунке 4.16 приведены оценки вероятности правильного распознавания для фиксированной частоты ложной тревоги $5 \cdot 10^{-4}$ всех обученных моделей СНС. В таблицах 9 – 12 приведены все оценки характеристик качества распознавания малоразмерных изображений СНС.

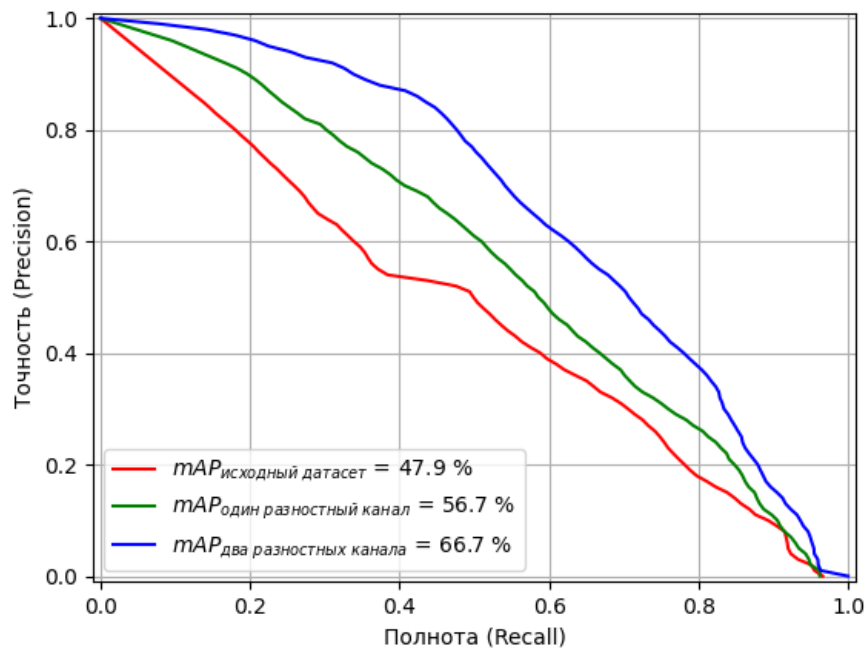


Рисунок 4.7 – Кривые $Precision - Recall$ для моделей СНС YOLOv3-Tiny

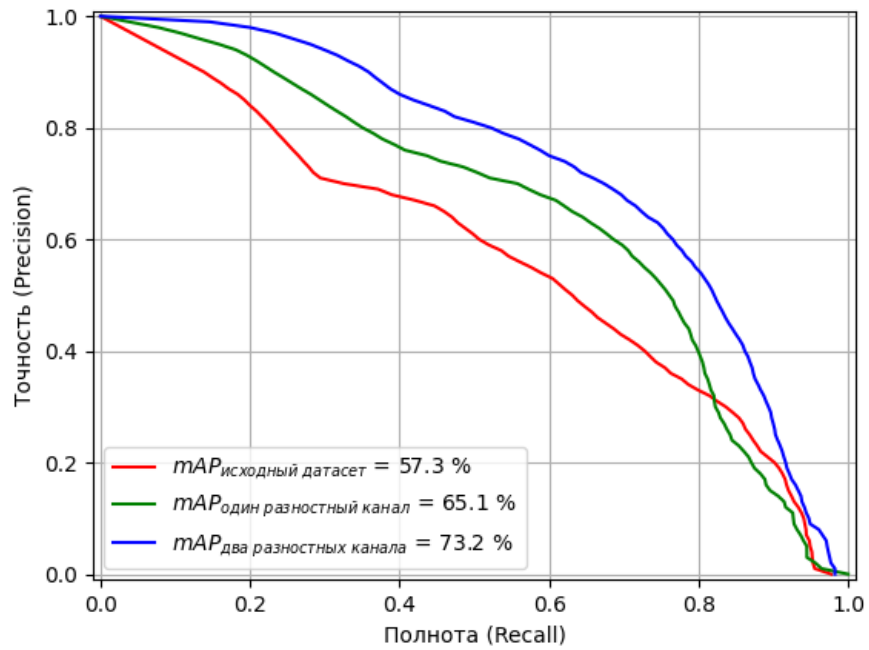


Рисунок 4.8 – Кривые *Precision – Recall* для моделей СНС YOLOv5s

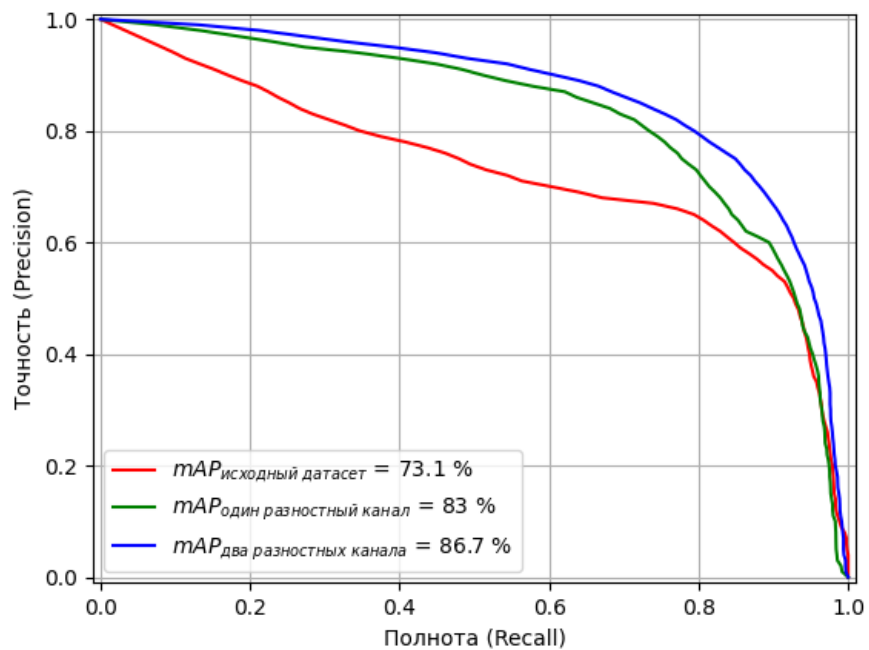


Рисунок 4.9 – Кривые *Precision – Recall* для моделей СНС YOLOv5l

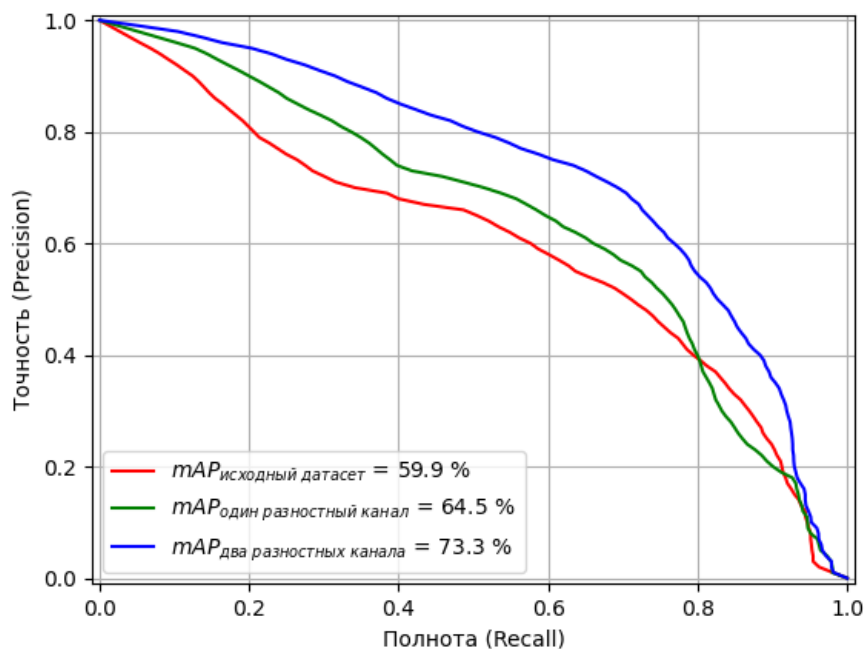


Рисунок 4.10 – Кривые *Precision – Recall* для моделей СНС YOLOv8s

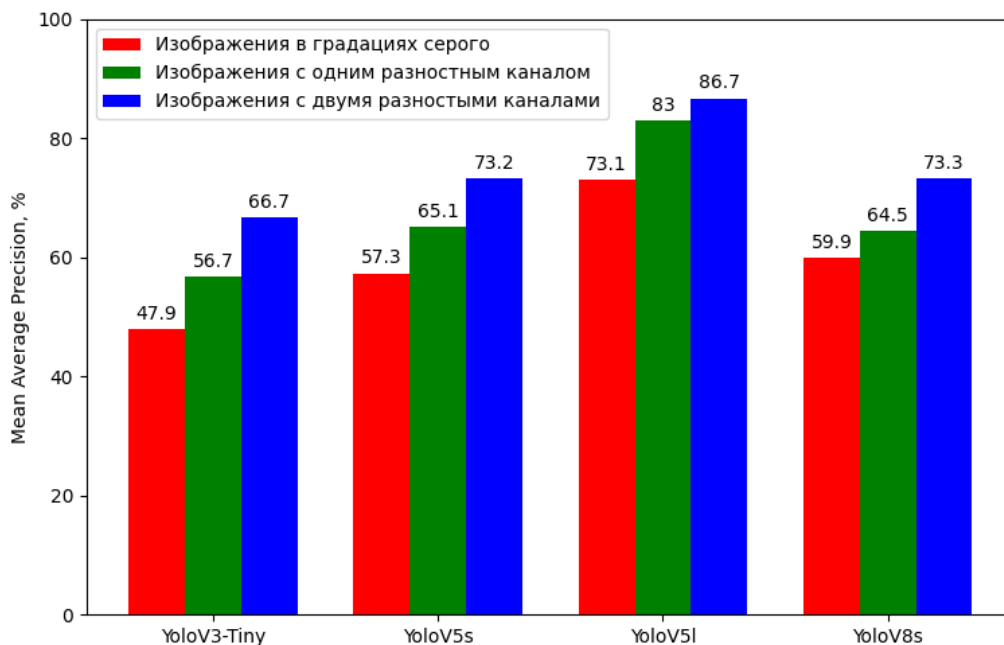


Рисунок 4.11 – Оценки mAP моделей СНС, обученных на датасетах: красный цвет – изображения в градациях серого; зелёный цвет – комплексированные изображения с одним разностным каналом; синий цвет – комплексированные изображения с двумя разностными каналами

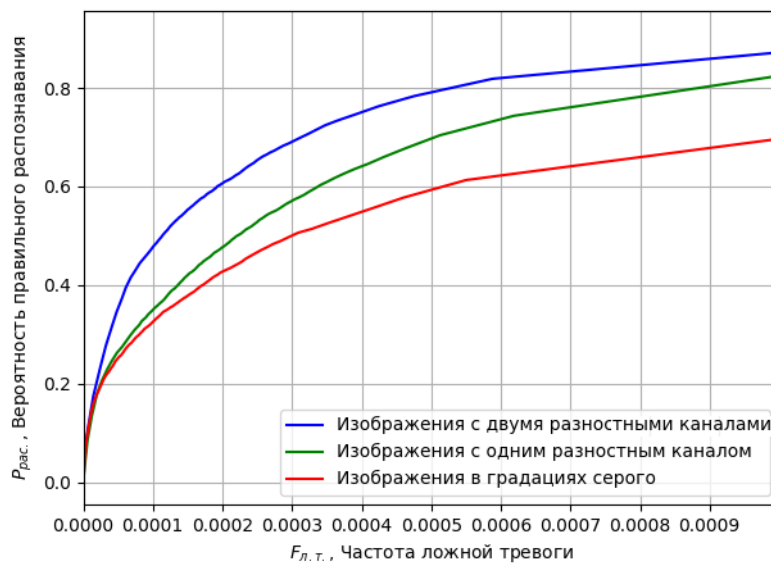


Рисунок 4.12 – Зависимость вероятности правильного распознавания от частоты ложной тревоги для моделей CHC YOLOv3-Tiny

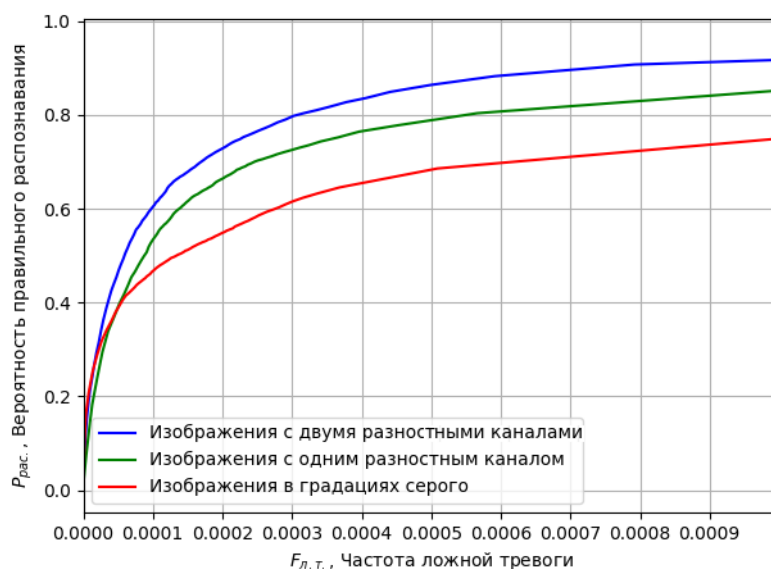


Рисунок 4.13 – Зависимость вероятности правильного распознавания от частоты ложной тревоги для моделей CHC YOLOv5s

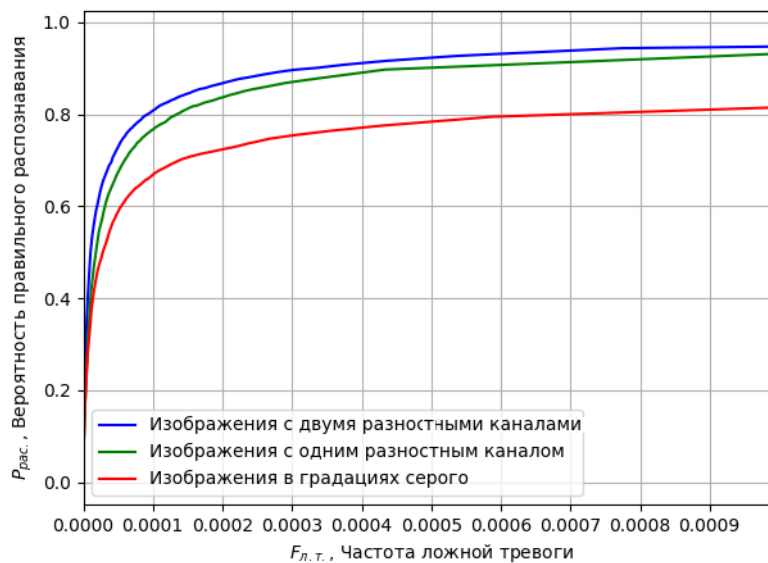


Рисунок 4.14 – Зависимость вероятности правильного распознавания от частоты ложной тревоги для моделей CHC YOLOv5l

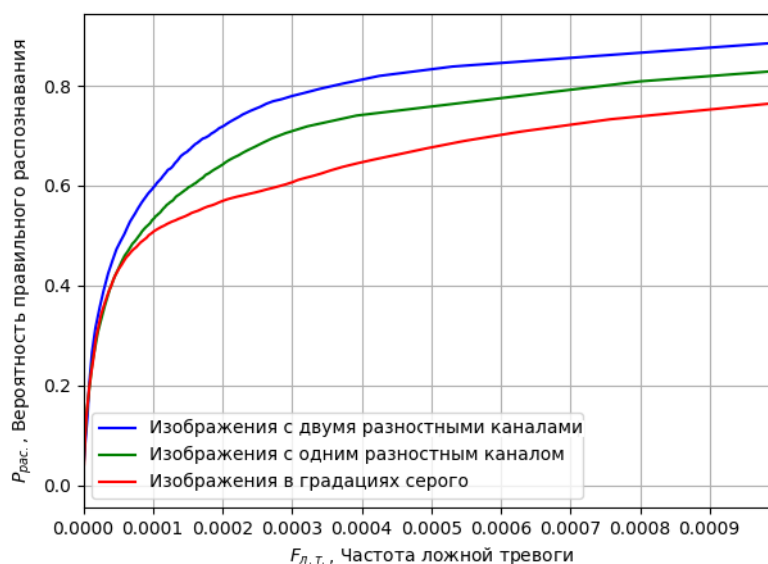


Рисунок 4.15 – Зависимость вероятности правильного распознавания от частоты ложной тревоги для моделей CHC YOLOv8s

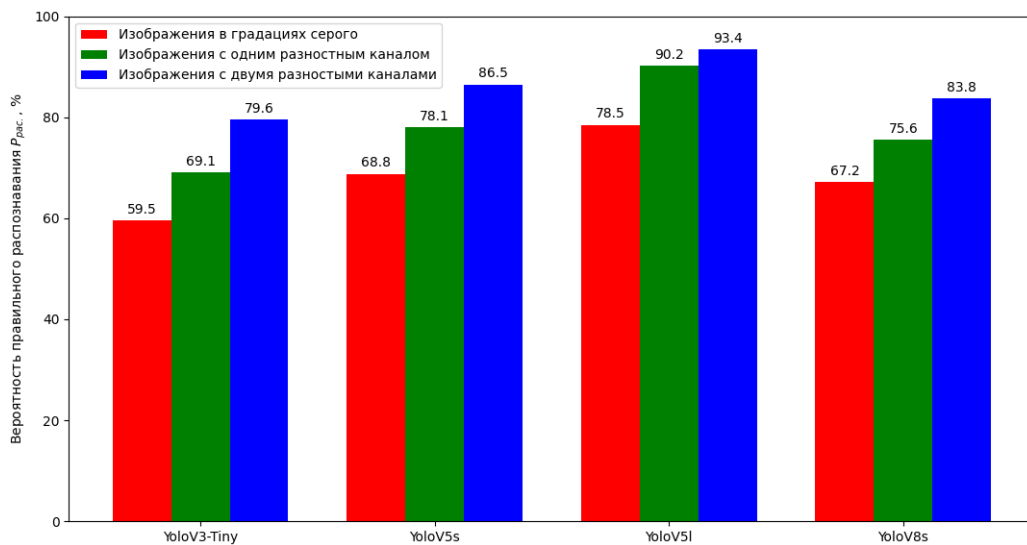


Рисунок 4.16 – Вероятности распознавания тестовых моделей СНС при фиксированной частоте ложной тревоги $F_{лт} = 5 \cdot 10^{-4}$, обученных на датасетах: красный цвет – изображения в градациях серого; зелёный цвет – комплексированные изображения с одним разностным каналом; синий цвет – комплексированные изображения с двумя разностными каналами

Таблица 9 – Характеристики качества распознавания для моделей СНС YOLOv3-Tiny

Датасет	$AP_0, \%$	$AP_1, \%$	$AP_2, \%$	$AP_3, \%$	$AP_4, \%$	$AP_5, \%$	$mAP, \%$	$P_{рас}, \%$
Изображения в градациях серого	61.2	53.2	44.2	51.6	49.2	27.7	47.9	59.5
Компл-ные изображения с одним разностным каналом	75.5	57.7	52.2	55.5	55.8	42.9	56.7	69.1
Компл-ные изображения с двумя разностными каналами	91.8	72.6	66.8	57.5	56.4	54.6	66.7	79.6

Таблица 10 – Характеристики качества распознавания для моделей СНС YOLOv5s

Датасет	AP_0 %	AP_1 %	AP_2 %	AP_3 %	AP_4 %	AP_5 %	mAP %	$P_{рас}$ %
Изображения в градациях серого	72.8	60.4	62.7	50.9	49.4	47.7	57.3	68.8
Компл-ные изображения с одним разностным каналом	75.2	75.7	66.7	53.1	65.2	54.5	65.1	78.1
Компл-ные изображения с двумя разностными каналами	79.2	79.2	71.6	59.9	75.4	74.1	73.2	86.5

Таблица 11 – Характеристики качества распознавания для моделей СНС YOLOv5l

Датасет	AP_0 %	AP_1 %	AP_2 %	AP_3 %	AP_4 %	AP_5 %	mAP %	$P_{рас}$ %
Изображения в градациях серого	69.8	81.7	75	72	74.1	66.2	73.1	78.5
Компл-ные изображения с одним разностным каналом	90.2	90.6	85.5	79.1	80.9	71.1	83	90.2
Компл-ные изображения с двумя разностными каналами	94.4	93	83.1	82.7	86.1	81.1	86.7	93.5

Таблица 12 – Характеристики качества распознавания для моделей СНС YOLOv8s

Датасет	AP_0 %	AP_1 %	AP_2 %	AP_3 %	AP_4 %	AP_5 %	mAP %	$P_{рас}$ %
Изображения в градациях серого	67.7	59	59.9	68.8	56.3	47.5	59.9	67.2
Компл-ные изображения с одним разностным каналом	72.7	68.6	69.7	57.3	70.7	48.2	64.5	75.6
Компл-ные изображения с двумя разностными каналами	77.2	79.4	67.9	52.6	86.8	75.9	73.3	83.8

СНС YOLOv3-TINY, обученная на комплексированных изображениях с двумя разностными каналами, повысила качество распознавания на 10 % mAP или 10.5 % $P_{рас}$, чем обученная на комплексированных изображениях с одним разностным каналом и на 18.8 % mAP или 20.1 % $P_{рас}$, чем обученная на изображениях в градациях серого. СНС YOLOv5s, обученная на комплексированных изображениях с двумя разностными каналами, повысила качество распознавания на 8.1 % mAP или 8.4 % $P_{рас}$, чем обученная на комплексированных изображениях с одним разностным каналом и на 15.9 % mAP или 17.7 % $P_{рас}$, чем обученная на изображениях в градациях серого. YOLOv5l, обученная на комплексированных изображениях с двумя разностными каналами, повысила качество распознавания на 3.7 % mAP или 3.3 % $P_{рас}$, чем обученная на комплексированных изображениях с одним разностным каналом и на 13.6 % mAP или 14.9 % $P_{рас}$, чем обученная на изображениях в градациях серого. YOLOv8s, обученная на комплексированных изображениях с двумя разностными каналами, повысила качество распознавания на 8.8 % mAP или 8.2 % $P_{рас}$, чем обученная на комплексированных изображениях с одним разностным каналом и на 13.4 % mAP или 16.7 % $P_{рас}$, чем обученная на изображениях в градациях серого. Анализ полученных результатов приводит к следующим выводам. Применение разработанного алгоритма предварительной обработки изображений позволило повысить качество распознавания малоразмерных изображений всех тестовых моделей СНС.

4.2 Вычислительный эксперимент с целью исследования разработанного алгоритма предварительной обработки изображений на нейронном ускорителе

4.2.1 Описание вычислительного эксперимента с целью исследования алгоритма предварительной обработки изображений на нейронном ускорителе

Разработанный алгоритм предварительной обработки изображений с целью комплексирования признаков движения формирует новый экстрактор признаков путём объединения алгоритма расширения признакового пространства с экстрактором признаков СНС. Прямое распространение данных через новый экстрактор признаков, в том числе и через алгоритм предварительной обработки изображений, проводится на вычислительных мощностях нейронного ускорителя. Таким образом, в экспериментальной части данной главы приведено сравнение эффективности предложенного алгоритма путём сравнения времени выполнения алгоритма на нейронном ускорителе и центральном процессоре.

Чтобы сравнить скорость выполнения, проведено построение алгоритма на центральном процессоре и нейронных ускорителях вычислительных устройств. Релевантность экспериментальных результатов достигается за счет оценки времени работы алгоритма на различных вычислительных устройствах NVIDIA Jetson Nano, NVIDIA Jetson TX2, NVIDIA Jetson AGX Xavier и Rockchip RK3588. Подробное описание технических характеристик вычислительных устройств рассмотрено в следующем пункте данной главы.

Построение алгоритма предварительной обработки изображений на вычислительных устройствах NVIDIA Jetson. Для построения алгоритма предварительной обработки изображений на устройствах линейки NVIDIA Jetson используется среда разработки JetPack, созданная для разработки ИИ. Среда разработки JetPack содержит пакет драйверов Linux4Tegra на базе операционной системы (ОС) Linux Ubuntu с библиотеками для программного ускорения об-

работки данных CUDA, CUDA-X и API-интерфейсами для глубокого обучения, компьютерного зрения, ускоренных вычислений [127, 133–139].

Для выполнения алгоритма на центральном процессоре используется CPU-модель алгоритма реализованная на языке программирования *C++*, которая достигла более высокой скорости обработки данных по сравнению с *Python* реализацией (листинг программных компонент представлен в приложении А.3).

Чтобы перенести вычисления алгоритма предварительной обработки изображений на вычислительные мощности графического процессора, необходимо реализовать GPU-модель алгоритма с помощью фреймворка *TensorRT*. Применение фреймворка *TensorRT* обусловлено тем, что он использует весь потенциал вычислительной мощности GPU благодаря ключевым особенностям:

1. Калибровка модели позволяет проводить вычисления модели с пониженной разрядностью FP16, INT8;
2. Слияние слоев – это процесс оптимизации, который объединяет несколько слоев сети в одну операцию;
3. Динамическое управление памятью, позволяет эффективно использовать память во время прямого вывода модели.

Построение GPU-модели алгоритма предварительной обработки изображений условно можно представить в виде структурной схемы, изображенной на рисунке 4.17 а. На основе фреймворка *Pytorch* строится модель алгоритма с расширением *.pt. *Pytorch* модель – преобразуется в формат *ONNX*. *ONNX* является унифицированным форматом хранения моделей ИС, который поддерживается большинством популярных фреймворков глубокого обучения. На основе *ONNX* модели формируется *Engine* модель алгоритма. Чтобы реализовать *Engine* модель алгоритма с разрядностью вычислений INT8, необходимо провести калибровку модели с помощью квантования весовых коэффициентов методом PQT.

Построение алгоритма предварительной обработки изображений на вычислительном устройстве Rickchip. В необходимый набор программных средств для построения NPU-модели алгоритма предварительной обработки изображений

на вычислительном устройстве Rockchip RK3588 входят ОС Linux Ubuntu SDK (Software Development Kit) Rockchip Neural Network-Toolkit2 (RKNN-Toolkit2) и RKNN-Toolkit-Lite2. RKNN-Toolkit2 – это комплект средств разработки программного обеспечения, позволяющий выполнять преобразование моделей, прямой вывод и оценку производительности на персональных компьютерах (ПК) и вычислительных устройствах Rockchip. RKNN-Toolkit-Lite2 предоставляет интерфейсы программирования *Python* для платформы Rockchip, позволяющий строить NPU-модели.

Построением NPU-модели СНС представлено в виде структурной схемы на рисунке 4.17 б. Как и в построении GPU-модели формируется NPU-модель алгоритма в унифицированном формате ONNX. На основе фреймворка RKNN, входящего в состав RKNN-Toolkit2, формируется RKNN модель алгоритма. Для калибровки RKNN-модели в целочисленных вычислениях INT8 используется метод статического квантования PQT. Импорт и запуск RKNN модели алгоритма осуществляется с помощью фреймворка RKNNLite, входящего в состав RKNN-Toolkit-Lite2.

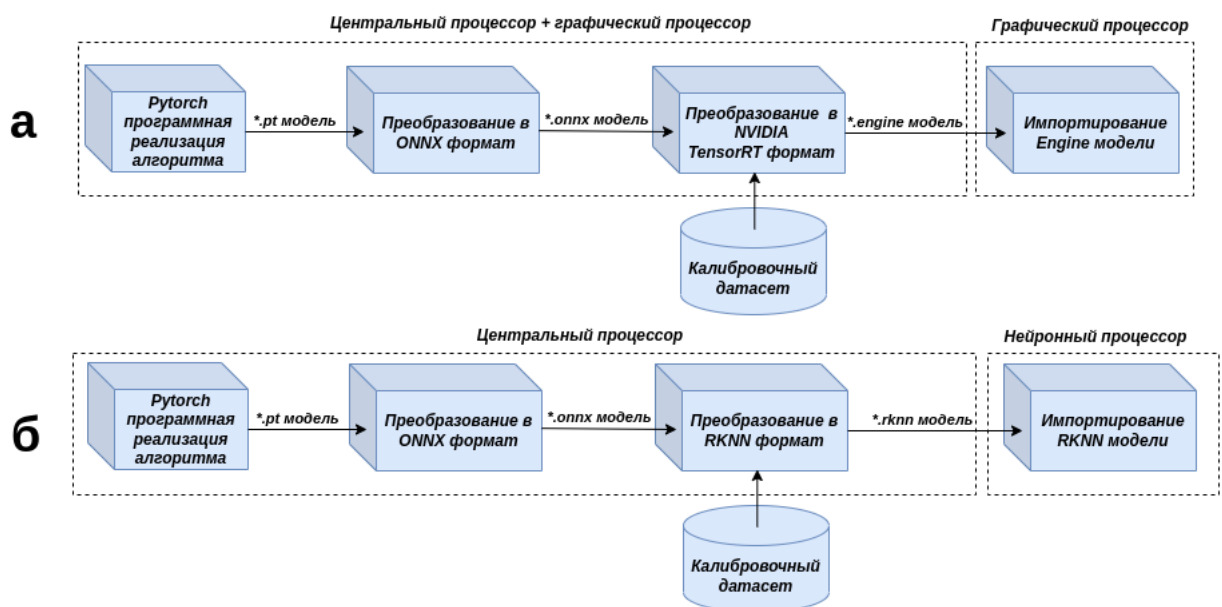


Рисунок 4.17 – Структурные схемы построения GPU и NPU моделей алгоритма предварительной обработки изображений на вычислительных устройствах NVIDIA Jetson и Rockchip

Далее рассмотрены вычислительные устройства, используемые в эксперименте.

4.2.2 Вычислительные устройства с нейронным ускорителем

Как правило, СНС строятся на встраиваемых устройствах с аппаратной поддержкой нейроускорителя [127, 133, 135, 136, 138]. В большинстве случаев в качестве нейроускорителя применяются графический и нейронный процессоры. Для проведения вычислительных экспериментов был выбран ряд вычислительных устройств, исходя из следующих критериев: 1) наличие поддержки нейронного ускорителя; 2) потребляемая мощность вычислительного устройства не превышает 30 Вт; 3) наличие гибкого ПО, позволяющего управлять вычислительными мощностями нейронного ускорителя. Среди вычислительных устройств с поддержкой графического процессора можно выделить линейку устройств Jetson от компании NVIDIA.

NVIDIA Jetson Nano – это бюджетный компактный компьютер, который обладает производительностью и энергоэффективностью, необходимой для современных задач ИИ, параллельной работы нескольких НС и одновременной обработки данных с различных сенсорных датчиков (рисунок 4.18). Вычислительный модуль оснащён центральным процессором Quad-core ARM Cortex-A57 MPCore processor и графическим процессором 128-core NVIDIA Maxwell architecture GPU. Jetson Nano представляет начальное решение для внедрения ИИ во встраиваемые системы. Модуль Jetson Nano обеспечивает до 472 Giga Operations per Second (GOPS), поддерживает различные режимы энергопотребления 5, 10 Вт. Вычислитель на аппаратном уровне оснащён тензорными ядрами, которые позволяют проводить вычисления с полной, половинной и смешанной разрядностью.

NVIDIA Jetson TX2 (8GB) обеспечивает вычислительные возможности для систем ИИ благодаря архитектуре графического процессора NVIDIA Pascal,



Рисунок 4.18 – Вычислительное устройство NVIDIA Jetson Nano

объёму памяти до 8 Гб (рисунок 4.19). Вычислительный модуль оснащён центральными процессорами Dual-core NVIDIA Denve 2 64-bit CPU and quad-core Arm Cortex-A57 MPCore processor и графическим процессором 256-core NVIDIA Pascal architecture GPU. Модуль серии Jetson TX2 может обеспечить обработку данных в масштабе реального времени, высокую пропускную способность. Jetson TX2 достигает до 1.33 Tera Operations per Second (TOPS), поддерживает различные режимы энергопотребления 7.5, 15 Вт. NVIDIA Jetson TX2 позволяет проводить вычисления с полной, половинной и смешанной разрядностью.

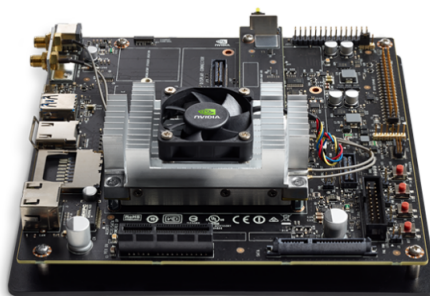


Рисунок 4.19 – Вычислительное устройство Jetson TX2

NVIDIA Jetson AGX Xavier – это компактный мощный модуль, который обеспечивает аппаратное ускорение прямого вывода нейросетевых алгоритмов и высокую скорость обработки данных (рисунок 4.20). Вычислительный модуль оснащён центральным процессором 8-core NVIDIA Carmel Armv8.2 64-bit CPU 8MB L2 + 4MB L3 и графическим процессором NVIDIA Volta architecture with 512 NVIDIA CUDA cores and 64 Tensor cores. Модуль Jetson AGX Xavier обеспе-

чивает до 32 TOPS, поддерживает различные режимы энергопотребления 10, 15 и 30 Вт. Jetson AGX Xavier обеспечивает устойчивость к широкому диапазону температур, ударов и вибраций. NVIDIA Jetson AGX Xavier на аппаратном уровне оснащен тензорными ядрами, которые позволяют проводить вычисления с полной, половинной, целочисленной и смешанной разрядностью.



Рисунок 4.20 – Вычислительное устройство NVIDIA Jetson AGX Xavier

Rockchip RK3588 – это высокопроизводительное устройство с низким энергопотреблением для встраиваемых вычислительных устройств на базе Advanced RISC Machine, объединенный четырехъядерным процессором, в состав которого входят процессоры Cortex-A76 и Cortex-A55 с отдельным сопроцессором NEON (рисунок 4.21). Встроенный нейронный процессор поддерживает вычисления с полной и целочисленной разрядностью, обладает производительностью до 6 TOPS. Максимальная потребляемая мощность 12 Вт. Кроме того, благодаря высокой совместимости фреймворков можно легко конвертировать модели ИС в необходимый формат RKNN из таких фреймворков как *TensorFlow*, *MXNet*, *PyTorch*, *Caffe* и др.

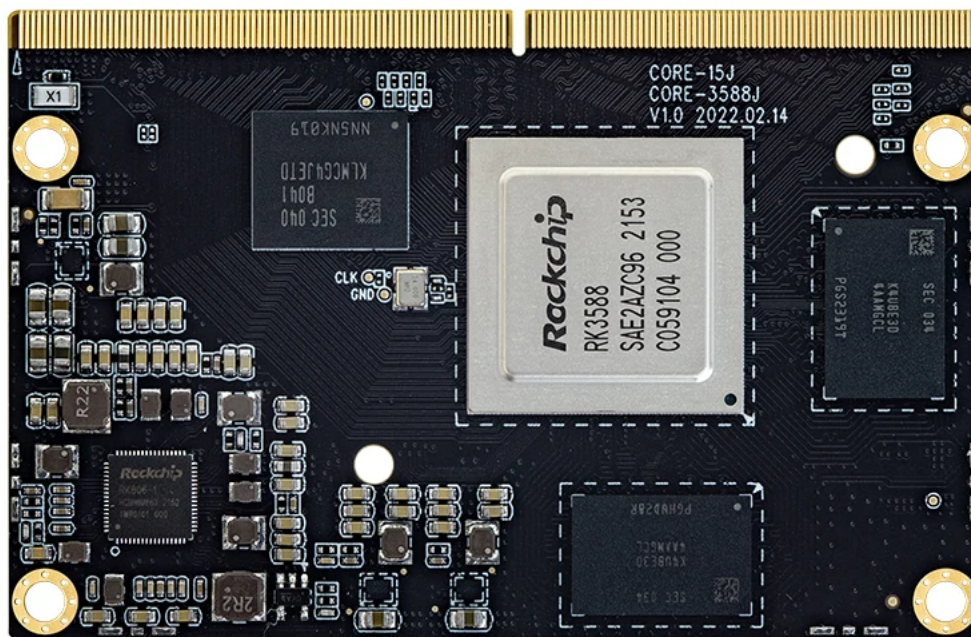


Рисунок 4.21 – Вычислительное устройство Rockchip RK3588

4.2.3 Оценка времени выполнения разработанного алгоритма предварительной обработки изображений на нейронном ускорителе

Оценка времени выполнения алгоритма предварительной обработки изображений на вычислительных устройствах проводилась в режиме максимально допустимой вычислительной мощности для всех устройств: устройства линейки Jetson в режиме *jetson_clocks*, Rockchip RK3588 режим загрузки всех вычислительных ядер нейронного процессора.

Оценка времени выполнения CPU-моделей алгоритма предварительной обработки изображений на центральном процессоре проводилась с помощью библиотеки *Chrono*.

Фреймворк *TensorRT* предоставляет возможность профилирования GPU-моделей послойно с помощью исполняемого файла *trtexec*. Профилирование – это сбор характеристик программы, таких как время выполнения, объём потребляемой памяти и др. Таким образом, можно оценить время выполнения алгоритма послойно и в целом. Оценка времени выполнения NPU-моделей осуществляется с помощью программного функционала *rknn.eval_perf()* от RKNN.

На каждом вычислительном устройстве построены модели алгоритма предварительной обработки изображений для всех поддерживаемых разрядностей вычислений. После чего проведена оценка времени выполнения алгоритма для размера входных данных 1 x 3 x 960 x 960, которая отображена в таблице 13.

Таблица 13 – Оценка времени выполнения алгоритма предварительной обработки изображений

Вычислительное устройство	Нейронный ускоритель	Разрядность вычислений	Время выполнения, мс
NVIDIA Jetson Nano B01	CPU	FP32	13.1
	GPU	FP32	8.9
		FP16	8.8
NVIDIA Jetson TX2 8GB	CPU	FP32	5.8
	GPU	FP32	3.6
		FP16	3.5
NVIDIA Jetson AGX Xavier	CPU	FP32	3.5
	GPU	FP32	1.2
		FP16	1.1
		INT8	1.1
Rockchip RK3588	CPU	FP32	3.7
	NPU	INT8	2.8

На основе полученных результатов проведённого вычислительного эксперимента можно сделать вывод, что скорость обработки данных алгоритма на нейронных ускорителях повысилась для всех разрядностей вычислений по сравнению с центральным процессором. В частности, прирост скорости обработки данных алгоритма на вычислительном устройстве NVIDIA Jetson Nano (B01) увеличился в 1.5 раза, NVIDIA Jetson TX2 увеличился в 1.7 раза, NVIDIA Jetson AGX Xavier увеличился в 3.2 раза, Rockchip RK3588 увеличился в 1.3 раза. Эффективность алгоритма предварительной обработки для нейронного ускорителя зависит от производительности вычислителя и архитектуры модели. Переход к вычислениям с пониженной разрядностью не привёл к увеличению скорости обработки данных из-за недостаточной глубины и количества слоев модели алгоритма пред-

варительной обработки данных, что нельзя сказать про глубокие детекционные модели СНС. Проведенное исследование по оценке времени выполнения детекционных СНС в научной работе [127] подтверждает, что переход к вычислениям с пониженной разрядностью приводит к увеличению скорости обработки данных, но в определённых случаях снижается качество распознавания [127]. Далее рассмотрено исследование влияния перехода к вычислениям с пониженной разрядностью на качество распознавания и время выполнения тестовых СНС на вычислительных устройствах.

4.2.4 Оценка качества распознавания и времени выполнения свёрточных нейронных сетей при переходе к вычислениям с пониженной разрядностью

Алгоритм предварительной обработки изображений с целью комплексирования признаков движения на нейронном ускорителе формирует новый экстрактор признаков. Для повышения скорости обработки входных данных экстрактором признаков применяется переход к вычислениям с пониженной разрядностью, но такой подход может приводить к снижению качества распознавания. Поэтому рассмотрим влияние перехода к вычислениям с пониженной разрядностью на качество распознавания тестовых СНС и время выполнения.

На основе обученных моделей СНС (таблица 3) на датасете изображений с двумя разностными каналами построены GPU и NPU модели с модифицированными экстракторами признаков для всех поддерживаемых разрядностей вычислений на основе фреймворков TensorRT и RKNN. В таблице 14 представлен список модифицированных архитектур.

Таблица 14 – Модифицированные архитектуры свёрточных нейронных сетей

Архитектура	Экстрактор признаков	Агрегатор признаков	Декодер признаков
YOLOv3-Tiny-M	алгоритм предварительной обработки изображений + Darknet19	FPN	YOLO
YOLOv5s-M	алгоритм предварительной обработки изображений + YOLOv5s	PAN	YOLO
YOLOv5l-M	алгоритм предварительной обработки изображений + YOLOv5l	PAN	YOLO
YOLOv8s-M	алгоритм предварительной обработки изображений + YOLOv8s	PAN	YOLO

Модели с целочисленной разрядностью вычислений получены с помощью калибровки методом PQT. Далее проведена оценка качества распознавания изображений объектов СНС на тестовом датасете с помощью характеристики mAP и вероятности правильного распознавания при фиксированной частоте ложной тревоги $F_{лт} = 5 \cdot 10^{-4}$, значения которых представлены в таблице 15. Оценки времени выполнения модифицированных моделей СНС на вычислительных устройствах приведены в таблице Б.

Оценочные значения характеристик качества распознавания тестовых моделей СНС приводят к следующим выводам: переход к половинной разрядности вычислений GPU-моделей не влияет на качества распознавания. Переход к целочисленной разрядности вычислений GPU-моделей приводит к снижению качества распознавания на 6.5 – 19.1 % mAP или 5.4 – 15.9 % $P_{рас}$. В свою очередь, переход к вычислениям с половинной разрядностью GPU-моделей позволил сократить время обработки данных в 1.6 – 3.3 раза, а для к целочисленной разрядности в 3.6 – 6 раза в зависимости от модели СНС.

Переход к целочисленной разрядности вычислений NPU-моделей приводит к снижению качества распознавания на 5 – 13.9 % mAP или 4.6 – 14.4 % $P_{рас}$ по сравнению с вычислениями FP32 на GPU.

Таблица 15 – Оценки качества распознавания тестовых моделей СНС для вычислений с различной разрядностью

Архитектура СНС	Тип нейронного ускорителя	Разрядность выч-ий	mAP, %	$P_{\text{рас}}$, %
YOLOv3-Tiny-M	GPU	FP32	66.7	79.6
		FP16	66.8	79.7
		INT8	49.7	63.1
	NPU	INT8	52.8	65.2
YOLOv5s-M	GPU	FP32	73.2	86.5
		FP16	73.3	86.4
		INT8	54.2	73.6
	NPU	INT8	63.4	76.1
YOLOv5l-M	GPU	FP32	86.7	93.4
		FP16	86.7	93.4
		INT8	80.3	87.8
	NPU	INT8	81.7	88.7
YOLOv8s-M	GPU	FP32	73.3	83.8
		FP16	73.2	83.7
		INT8	65.1	78.4
	NPU	INT8	64.3	79.2

4.3 Разработка интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом

4.3.1 Описание исходного состава компонент и требуемых характеристик распознающей системы

Согласно разработанному алгоритму построения интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом, необходимо задать требуемые характеристики системы: время выполнения алгоритма не более 60 мс, вероятность распознавания не менее 85 % при частоте ложной тревоги $5 \cdot 10^{-4}$, потребляемая мощность вычислительного устройства не превышает 30 Вт.

В вычислительном эксперименте используются устройства потребляемая, мощность которых не превышает 30 Вт. Вычислительные устройства оснащены нейронными ускорителями типа GPU или NPU. С поддержкой графического процессора устройства от NVIDIA: Jetson Nano B01, Jetson TX2, Jetson AGX Xavier, а также устройство с поддержкой нейронного процессора RK3588 от компании Rockchip.

Для повышения качества распознавания малоразмерных изображений объектов используется разработанный алгоритм предварительной обработки изображений (рисунок 2.10). С учётом того, что вычислители оснащены нейронным ускорителем, то для сокращения времени обработки данных целесообразно перенести алгоритм предварительной обработки изображений на вычислительные мощности нейронного ускорителя, согласно разработанному алгоритму, структурная схема которого представлена на рисунке 3.1.

В качестве обучающих и тренировочных датасетов используются синтетические изображения с двумя разностными каналами, сгенерированными с помощью разработанного алгоритма формирования аннотированных условно-реальных разномодальных изображений. Далее представлен ход действий построения интеллектуальной системы, удовлетворяющей заданным характеристикам.

4.3.2 Построение интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом

В данной части главы приводится последовательность действий, направленная на создание интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом согласно разработанному алгоритму.

- 1. Формирование исходного состава компонент и требуемых характеристик.*

а) *Требуемые характеристики.* Требуемые характеристики распознающей системы должны соответствовать следующим критериям: потребляемая мощность не более 30 Вт, вероятность распознавания не менее 85 % при частоте ложной тревоги $5 \cdot 10^{-4}$, время выполнения не более 60 мс.

б) *Формирование множества вычислительных устройств.* Сформировано множество устройств с ограниченным вычислительным ресурсом, потребляемая мощность которых не превышает 30 Вт (таблица 16).

Таблица 16 – Вычислительные устройства

№	Вычислительное устройство	Потребляемая мощность, Вт
1	NVIDIA Jetson Nano B01	10
2	Rockchip RK3588	12
3	NVIDIA Jetson TX2 (8GB)	15
4	NVIDIA Jetson AGX Xavier	30

в) *Формирование множества архитектур СНС.* Сформировано множество одноэтапных архитектур СНС с различной вычислительной сложностью (таблица 3). Алгоритм распознавания построен согласно структурной схемы, представленной на рисунке 3.1. В таблице 14 представлено множество архитектур с модифицированным экстрактором признаков.

г) *Формирование данных.* В качестве тренировочных и тестовых выборок используются датасеты изображений с двумя разностными каналами. В качестве калибровочного датасета используется тестовая выборка.

2. *Формирование наборов.* На основе сформированного состава компонент производится формирование наборов, состоящих из вычислительного устройства с программной поддержкой, модели СНС, разрядности вычислений и данных. С помощью полученных наборов построены распознающие системы, после чего проведена оценка характеристик системы. Характеристики систем, удовлетворяющие необходимым критериям, буферизуются (Приложение Б). Зачёркнутые значения означают, что данный набор не удовлетворяет пороговому значению

критерия. Таким образом, построено четыре системы, удовлетворяющие заданным требованиям, наборы которых состоят:

1) из вычислительного устройства NVIDIA Jetson AGX Xavier, фреймворка TensorRT, модифицированной архитектуры YOLOv5s-M, разрядности вычислений FP16, датасетов изображений с двумя разностными каналами;

2) вычислительного устройства NVIDIA Jetson AGX Xavier, фреймворка TensorRT, модифицированной архитектуры YOLOv5s-M, разрядности вычислений FP32, датасетов изображений с двумя разностными каналами;

3) вычислительного устройства NVIDIA Jetson AGX Xavier, фреймворка TensorRT, модифицированной архитектуры YOLOv5l-M, разрядности вычислений FP16, датасетов изображений с двумя разностными каналами;

4) вычислительного устройства NVIDIA Jetson AGX Xavier, фреймворка TensorRT, модифицированной архитектуры YOLOv5l-M, разрядности вычислений INT8, датасетов изображений с двумя разностными каналами.

3. *Выбор набора.* На данном этапе проводится выбор набора, с помощью которого можно построить распознающую систему, обеспечивающую лучшие требуемые характеристики. Для выбора лучшего набора рассчитаны обобщенные показатели значимости F по формуле (3.4), значения которых приведены в приложение Б. Показатели значимости F рассчитываются методом взвешенной суммы неоднородных нормированных критериев. Каждому критерию присвоен свой коэффициент значимости, равный $\omega_1 = 0.2$, $\omega_2 = 0.4$, $\omega_3 = 0.4$, где ω_1 – коэффициент значимости потребляемой мощности устройства, ω_2 – коэффициент значимости времени выполнения, ω_3 – коэффициент значимости качества распознавания. По результатам расчётов наибольшее значение показателя значимости 0.98 (подчёркнутое значение) набрал набор, состоящий из вычислительного устройства NVIDIA Jetson AGX Xavier, фреймворка TensorRT, архитектуры YOLOv5s-M, половинной разрядности вычислений FP16, датасетов изображений с двумя разностными каналами [129]. На рисунке 4.22 представлена структурная схема алгоритма построения распознающей системы на основе сформированного набора.

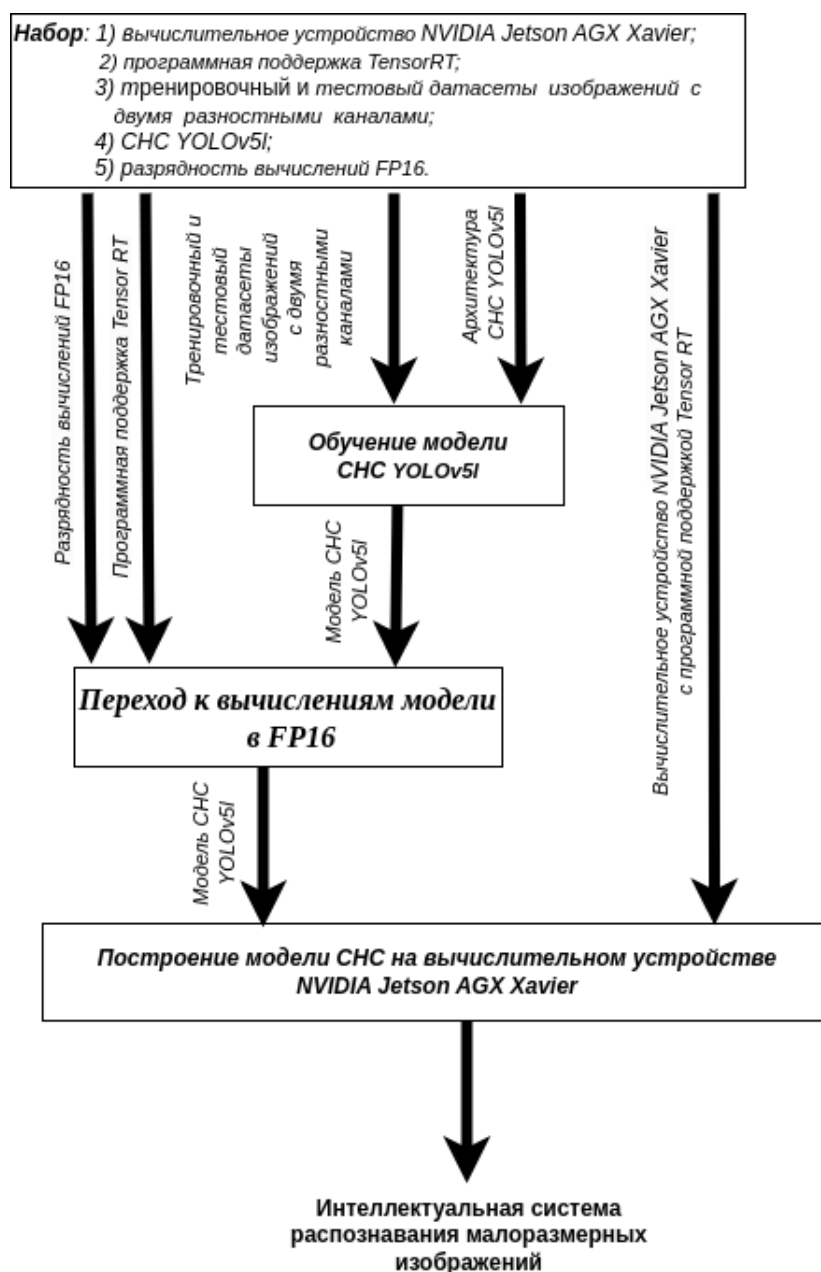


Рисунок 4.22 – Алгоритм построения созданной интеллектуальной системы распознавания малоразмерных изображений объектов

Построенная интеллектуальная система распознавания малоразмерных изображений объектов построена на базе разработанного алгоритма, который включает все разработанные методы и алгоритмы и покрывает весь процесс разработки от выбора архитектуры СНС, формирования датасетов, метода и алгоритма повышения качества распознавания и до выбора вычислительного устройства с применением методов оптимизации времени обработки данных, что вносит существенный вклад для развития методов и алгоритмов искусственного интеллекта и машинного обучения.

4.4 Выводы по материалам четвёртой главы

На основе разработанного алгоритма формирования аннотированных условно-реальных изображений сгенерированы датасеты для задачи распознавания малоразмерных изображений объектов: датасет изображений в градациях серого, датасет комплексированных изображений с одним и двумя разностными каналами.

Проведено обучение одноэтапных тестовых архитектур свёрточных нейронных сетей с различной вычислительной сложностью на тренировочных выборках. Частота ошибок распознавания СНС на датасете комплексированных изображений с двумя разностными каналами уменьшилась на 11.5 – 14.5 %. Таким образом, применение метода расширения признакового пространства признакам движения, полученными в процессе продолжительного наблюдения за объектом, позволило сформировать изображения, которые содержат больше различительных признаков.

Проведено исследование влияния разработанного алгоритма предварительной обработки изображений с целью комплексирования признаков движения на качество распознавания малоразмерных изображений объектов с помощью вычислительного эксперимента. По результатам проведенного вычислительного эксперимента можно сделать вывод, что свёрточные нейронные сети, обученные на изображениях с двумя разностными каналами, повысили вероятность распознавания на 14.9 – 20.1 %. Эти результаты говорят о том, что признаковое пространство изображений с двумя разностными каналами представлено более информативной системой признаков по сравнению с исходными изображениями. Таким образом, увеличение количества обрабатываемых изображений алгоритмом предварительной обработки сформировало комплексированное изображение, содержащее признаки движения, обладающие большей различительной способностью.

Проведён вычислительный эксперимент с целью исследования временных характеристик разработанного алгоритма предварительной обработки изображений на нейронном ускорителе. Исходя из полученных оценок времени выполнения алгоритма, можно сделать выводы, что выполнение алгоритма на нейронном ускорителе позволило повысить скорость обработки данных в 1.3 – 3.2 раза в зависимости от производительности и типа нейронного ускорителя.

Проведена апробация разработанного алгоритма построения системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом. Сформирован набор из вычислительного устройства NVIDIA Jetson AGX Xavier, фреймворка TensorRT, модифицированной архитектуры свёрточной нейронной сети YOLOv5s-M, половинной разрядности вычислений и датасетов изображений с двумя разностными каналами. С помощью определенного порядка действий над составными частями набора построена интеллектуальная система, удовлетворяющая вероятности распознавания 86.4 % при фиксированной частоте ложной тревоги $5 \cdot 10^{-4}$, времени выполнения 22 мс, потребляемой мощности устройства не более 30 Вт.

Заключение

В ходе выполнения диссертационного исследования достигнута поставленная цель, заключающаяся в повышении качества распознавания малоразмерных изображений объектов свёрточными нейронными сетями на устройствах с ограниченным вычислительным ресурсом. Научные и практические результаты диссертационного исследования имеют большое значение для развития технологий искусственного интеллекта при обработке и анализе изображений, успешно апробированы на конференциях и симпозиуме, а также внедрены в научно-исследовательской работе и учебный процесс.

Основные научные и практические результаты диссертационного исследования заключаются в следующем:

1. Разработан метод расширения признакового пространства с целью повышения информативности системы признаков. Расширение признакового пространства осуществляется с помощью предварительной обработки изображений, которая выделяет признаки движения и комплексирования с исходными признаками изображения. В расширенном признаковом пространстве минимизированы зоны неоднозначности за счёт увеличения расстояния между границами образов.
2. Разработан алгоритм предварительной обработки изображений с целью комплексирования признаков движения. Алгоритм формирует комплексированное изображение, в котором учтены исходные признаки изображения с признаками движения, полученными в процессе продолжительного наблюдения. Результирующее изображение является псевдоцветным изображением с цветными кластерами, характеризующими признаки движения объектов.
3. Для минимизации стоимости разработки распознающей системы на этапе обучения свёрточных нейронных сетей разработан алгоритм формирования условно-реальных разномодальных изображений, который

позволяет в автоматическом режиме генерировать аннотированные изображения, что позволяет уменьшить стоимость процедуры формирования датасетов.

4. Разработан алгоритм предварительной обработки изображений с целью комплексирования признаков движения на нейронном ускорителе. Алгоритм предварительной обработки изображений построен в рамках операций, поддерживаемых нейронными ускорителями, и является составной частью экстрактора признаков свёрточной нейронной сети, что позволяет обрабатывать данные на вычислительных мощностях нейронного ускорителя.
5. Разработан алгоритм построения системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом. Алгоритм формирует набор из вычислительного устройства с программной поддержкой, архитектуры свёрточной нейронной сети, датасетов и разрядности вычислений нейронного ускорителя. С помощью определенного порядка действий над составными частями набора можно построить распознающую систему, удовлетворяющую требованиям в качестве распознавания изображений объектов, времени обработки данных и допустимой потребляемой мощности.
6. Для исследования и апробации разработанного метода и алгоритмов сформированы датасеты аннотированных синтетических изображений для задачи распознавания малоразмерных изображений объектов с помощью разработанного алгоритма формирования аннотированных условно-реальных разномодальных изображений. Первый датасет состоит из исходных изображений в градациях серого, второй датасет состоит из комплексированных изображений с одним разностным каналом, третий состоит из комплексированных изображений с двумя разностными каналами.
7. Проведено исследование разработанного метода расширения признакового пространства с целью повышения информативности системы

признаков с помощью вычислительного эксперимента. Сформировано множество тестовых архитектур свёрточных нейронных сетей для задачи распознавания малоразмерных изображений объектов. Применение метода позволило уменьшить частоту ошибок распознавания на 11.5 – 14.5 %. Полученные значения частоты ошибок распознавания говорят о том, что применение метода расширения признакового пространства позволяет сформировать изображения, которые содержат больше различных признаков.

8. Проведено исследование влияния разработанного алгоритма предварительной обработки изображений на качество распознавания свёрточных нейронных сетей с помощью вычислительного эксперимента. Применение алгоритма позволило повысить вероятность распознавания свёрточных нейронных сетей на 14.9 – 20.1 %. Повышение качества распознавания обусловлено большей информативностью системы признаков, содержащихся в комплексированных изображениях с двумя разностными каналами.
9. Проведено исследование времени выполнения разработанного алгоритма предварительной обработки изображений для нейронного ускорителя с помощью вычислительного эксперимента. На основе полученных оценок времени выполнения можно сделать выводы, что применение предложенного подхода позволило сократить время выполнения алгоритма в 1.3 – 3.2 раза по сравнению с центральным процессором без изменения качества распознавания.
10. Создана интеллектуальная распознающая система, которая удовлетворяет заданным характеристикам в качестве распознавания, времени обработки данных и допустимой потребляемой мощности вычислительного устройства. Система построена на основе набора, состоящего: из вычислительного устройства NVIDIA Jetson AGX Xavier, фреймворка TensorRT, модифицированной архитектуры свёрточной нейронной сети YOLOv5s-M, половинной разрядности вычислений графического

процессора и датасетов комплексированных изображений с двумя разностными каналами. Система обеспечивает вероятность распознавания 86.4 % при фиксированной частоте ложной тревоги, время выполнения 22 мс и потребляемую мощность, не превышающую 30 Вт.

Все основные результаты диссертационного исследования направлены на решение научной задачи, достижения цели и являются значимыми для развития технологий искусственного интеллекта и машинного обучения.

Согласованность практических результатов, полученных в результате вычислительного эксперимента, с теоретическими предположениями и известными фактами теории распознавания образов подтверждает достоверность научных результатов.

Дальнейшим направлением исследований является разработка и анализ новых алгоритмов предварительной обработки последовательности кадров, позволяющих сформировать комплексированные изображения с большим количеством различительных признаков движения, для повышения качества распознавания малоразмерных изображений объектов алгоритмами искусственного интеллекта и машинного обучения.

Список сокращений и условных обозначений

AP	Average Precision
CIoU	Complete Intersection over Union
CmBN	Cross mini-Batch Normalization
CPU	Central Processing Unit
CSP	Cross-Stage-Partial-connections
CUDA	Compute Unified Device Architecture
cuDNN	Compute Unified Deep Neural Network
DETR	Detection Transformer
DPM	Deformable Part-based Model
FLOPS	Floating-Point Operations per Second
FN	False Negative
FP	False Positive
FP16	Floating Point 16
FP32	Floating Point 32
FPGA	Field Programmable Gate Array
FPN	Feature Pyramid Network
GIoU	Generalized Intersection over Union
GOPS	Giga Operations per Second
GPU	Graphics Processing Unit
HOG	Histogram of Oriented Gradients
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
INT8	Integer 8
IOU	Intersection Over Union
LSTM	Long Short-Term Memory
mAP	Mean Average Precision
NMS	Non Maximum Suppression

NPU	Neural Processing Unit
ONNX	Open Neural Network Exchange
PAN	Path Aggregation Network
PTQ	Post-Training Quantization
QAT	Quantization-aware Training
RKNN	Rockchip Neural Network
SAHI	Slicing Aided Hyper Inference
SAT	Self-Adversarial-Training
SDK	Software Development Kit
SPP	Spatial Pyramid Pooling
SSD	Single Shot Multibox Detector
TN	True Negative
TOPS	Tera Operations per Second
TP	True Positive
TPU	Tensor Processing Unit
VOC	Pattern Analysis, Statistical Modeling and Computational Learning Visual Object Classes
YOLO	You Only Look Once
ИИ	Искусственный Интеллект
ИК	Инфракрасный
НС	Нейронная Сеть
ОС	Операционная Система
ПК	Персональный Компьютер
ПЛИС	Программируемая Логическая Интегральная Схема
ПО	Программное Обеспечение
СНС	Свёрточная Нейронная Сеть

Список литературы

1. Методы автоматического обнаружения и сопровождения объектов. Обработка изображений и управление [Текст] / Б. Алпатов [и др.] // М.: Радиотехника. – 2008. – С. 176.
2. *Viola P., Jones M.* Rapid object detection using a boosted cascade of simple features [Текст] // Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001. Т. 1. – Ieee. 2001. – С. I–I.
3. *Viola P., Jones M. J.* Robust real-time face detection [Текст] // International journal of computer vision. – 2004. – Т. 57. – С. 137–154.
4. *Dalal N., Triggs B.* Histograms of oriented gradients for human detection [Текст] // 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). Т. 1. – Ieee. 2005. – С. 886–893.
5. *Felzenszwalb P., McAllester D., Ramanan D.* A discriminatively trained, multiscale, deformable part model [Текст] // 2008 IEEE conference on computer vision and pattern recognition. – Ieee. 2008. – С. 1–8.
6. R-CNN for small object detection [Текст] / С. Chen [и др.] // Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20–24, 2016, Revised Selected Papers, Part V 13. – Springer. 2017. – С. 214–230.
7. *Bochkovskiy A., Wang C.-Y., Liao H.-Y. M.* Yolov4: Optimal speed and accuracy of object detection [Текст] // arXiv preprint arXiv:2004.10934. – 2020.
8. Small object detection with multiscale features [Текст] / G. X. Hu [и др.] // International Journal of Digital Multimedia Broadcasting. – 2018. – Т. 2018.

9. An evaluation of deep learning methods for small object detection [Текст] / N.-D. Nguyen [и др.] // Journal of electrical and computer engineering. – 2020. – Т. 2020. – С. 1–18.
10. Small object detection in optical remote sensing video with motion guided R-CNN [Текст] / J. Feng [и др.] // IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium. – IEEE. 2020. – С. 272–275.
11. Feature-fused SSD: Fast detection for small objects [Текст] / G. Cao [и др.] // Ninth international conference on graphic and image processing (ICGIP 2017). Т. 10615. – SPIE. 2018. – С. 381–388.
12. *Zhang X., Izquierdo E., Chandramouli K.* Dense and small object detection in uav vision based on cascade network [Текст] // Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. – 2019. – С. 118–126.
13. Augmentation for small object detection [Текст] / M. Kisantal [и др.] // arXiv preprint arXiv:1902.07296. – 2019.
14. *Redmon J., Farhadi A.* Yolov3: An incremental improvement [Текст] // arXiv preprint arXiv:1804.02767. – 2018.
15. SSD: Single shot multibox detector [Текст] / W. Liu [и др.] // Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. – Springer. 2016. – С. 21–37.
16. *Zhou X., Wang D., Krähenbühl P.* Objects as points [Текст] // arXiv preprint arXiv:1904.07850. – 2019.
17. *Akyon F. C., Altinuc S. O., Temizel A.* Slicing aided hyper inference and fine-tuning for small object detection [Текст] // 2022 IEEE International Conference on Image Processing (ICIP). – IEEE. 2022. – С. 966–970.
18. Fcos: Fully convolutional one-stage object detection [Текст] / Z. Tian [и др.] // Proceedings of the IEEE/CVF international conference on computer vision. – 2019. – С. 9627–9636.

19. Multimodal medical supervised image fusion method by CNN [Текст] / Y. Li [и др.] // *Frontiers in Neuroscience*. – 2021. – Т. 15. – С. 638976.
20. Semi-hidden target recognition in gated viewer images fused with thermal IR images [Текст] / M. A. Smeelen [и др.] // *Information Fusion*. – 2014. – Т. 18. – С. 131–147.
21. Airborne infrared and visible image fusion for target perception based on target region segmentation and discrete wavelet transform [Текст] / Y. Niu [и др.] // *Mathematical Problems in Engineering*. – 2012. – Т. 2012.
22. *Liu S., Liu Z.* Multi-channel CNN-based object detection for enhanced situation awareness [Текст] // arXiv preprint arXiv:1712.00075. – 2017.
23. Moving object detection based on background compensation and deep learning [Текст] / J. Zhu [и др.] // *Symmetry*. – 2020. – Т. 12, № 12. – С. 1965.
24. Указ Президента РФ от 07.07. 2011 N 899 (ред. от 16.12. 2015) Об утверждении приоритетных направлений развития науки, технологий и техники в Российской Федерации и перечня критических технологий Российской Федерации [Электронный ресурс]. – URL: https://www.consultant.ru/document/cons_doc_LAW_116178/594d7dd112d737966851b3611b36f8cf3b09034c/ (дата обр. 18.02.2025).
25. *Загоруйко Н. Г.* Методы распознавания и их применение [Текст]. – 1972.
26. *Родионов С.* Основы оптики [Текст] // СПб: СПб ГИТМО (ТУ). – 2000.
27. Обработка и анализ изображений в задачах машинного зрения [Текст] / Ю. В. Визильтер [и др.]. – 2010.
28. *Вапник В., Червоненкис А.* Теория распознавания образов. Статистические проблемы обучения [Текст]. – 1974.
29. *Фомин Я. А., Тарловский Г. Р.* Статистическая теория распознавания образов [Текст]. – Радио и связь, 1986.

30. *Загоруйко Н.* Классификация задач распознавания образов [Текст] // Вычислительные системы. – 1966. – № 22. – С. 146–153.
31. *Дуда Р.* Распознавание образов и анализ сцен [Текст]. – Рипол Классик, 1976.
32. *Альков П.* Современные тенденции и направления развития классической теории распознавания в ик диапазоне [Текст] // Молодежный научно-технический вестник. – 2014. – № 1. – С. 29–29.
33. The targeting task performance (TTP) metric: a new model for predicting target acquisition performance [Текст] / R. H. Vollmerhausen [и др.] // US Army CERDEC, Fort Belvoir, VA. – 2004. – Т. 22060.
34. Метрики оценки алгоритмов автоматического сопровождения [Текст] / А. Е. Щелкунов, В. В. Ковалев, К. И. Морев, И. В. Сидько // Известия Южного федерального университета. Технические науки. – 2020. – 1 (211). – С. 233–245.
35. Generalized intersection over union: A metric and a loss for bounding box regression [Текст] / H. Rezatofighi [и др.] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. – 2019. – С. 658–666.
36. *Вентцель Е. С.* Теория вероятностей [Текст]. – «Юстиция», 2018.
37. *Вентцель Е., Овчаров Л.* Прикладные задачи теории вероятностей [Текст]. – Радио и связь, 1983.
38. The pascal visual object classes (voc) challenge [Текст] / M. Everingham [и др.] // International journal of computer vision. – 2010. – Т. 88. – С. 303–338.
39. *Padilla R., Netto S. L., Da Silva E. A.* A survey on performance metrics for object-detection algorithms [Текст] // 2020 international conference on systems, signals and image processing (IWSSIP). – IEEE. 2020. – С. 237–242.

40. *Krizhevsky A., Sutskever I., Hinton G. E.* Imagenet classification with deep convolutional neural networks [Текст] // *Advances in neural information processing systems*. – 2012. – Т. 25.
41. Imagenet large scale visual recognition challenge [Текст] / *O. Russakovsky [и др.]* // *International journal of computer vision*. – 2015. – Т. 115. – С. 211–252.
42. *Жерон О.* Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow: концепции, инструменты и техники для создания интеллектуальных систем. Пер. с англ [Текст] // *СПб.: ООО "Альфа-книга*. – 2018.
43. Efficient backprop [Текст] / *Y. LeCun [и др.]* // *Neural networks: Tricks of the trade*. – Springer, 2002. – С. 9–50.
44. *Mishkin D., Matas J.* All you need is a good init [Текст] // *arXiv preprint arXiv:1511.06422*. – 2015.
45. *Simonyan K., Zisserman A.* Very deep convolutional networks for large-scale image recognition [Текст] // *arXiv preprint arXiv:1409.1556*. – 2014.
46. Deep residual learning for image recognition [Текст] / *К. Хе [и др.]* // *Proceedings of the IEEE conference on computer vision and pattern recognition*. – 2016. – С. 770–778.
47. Aggregated residual transformations for deep neural networks [Текст] / *S. Xie [и др.]* // *Proceedings of the IEEE conference on computer vision and pattern recognition*. – 2017. – С. 1492–1500.
48. Densely connected convolutional networks [Текст] / *G. Huang [и др.]* // *Proceedings of the IEEE conference on computer vision and pattern recognition*. – 2017. – С. 4700–4708.
49. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size [Текст] / *F. N. Iandola [и др.]* // *arXiv preprint arXiv:1602.07360*. – 2016.

50. Mobilenets: Efficient convolutional neural networks for mobile vision applications [Текст] / A. G. Howard [и др.] // arXiv preprint arXiv:1704.04861. – 2017.
51. Searching for mobilenetv3 [Текст] / A. Howard [и др.] // Proceedings of the IEEE/CVF international conference on computer vision. – 2019. – С. 1314–1324.
52. Mobilenetv2: Inverted residuals and linear bottlenecks [Текст] / M. Sandler [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2018. – С. 4510–4520.
53. Mnasnet: Platform-aware neural architecture search for mobile [Текст] / M. Tan [и др.] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. – 2019. – С. 2820–2828.
54. Shufflenet: An extremely efficient convolutional neural network for mobile devices [Текст] / X. Zhang [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2018. – С. 6848–6856.
55. Shufflenet v2: Practical guidelines for efficient cnn architecture design [Текст] / N. Ma [и др.] // Proceedings of the European conference on computer vision (ECCV). – 2018. – С. 116–131.
56. Feature pyramid networks for object detection [Текст] / T.-Y. Lin [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2017. – С. 2117–2125.
57. Path aggregation network for instance segmentation [Текст] / S. Liu [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2018. – С. 8759–8768.
58. *Tan M., Pang R., Le Q. V.* Efficientdet: Scalable and efficient object detection [Текст] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. – 2020. – С. 10781–10790.

59. *Ghiasi G., Lin T.-Y., Le Q. V.* Nas-fpn: Learning scalable feature pyramid architecture for object detection [Текст] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. – 2019. – С. 7036–7045.
60. *Girshick R.* Fast r-cnn [Текст] // Proceedings of the IEEE international conference on computer vision. – 2015. – С. 1440–1448.
61. Rich feature hierarchies for accurate object detection and semantic segmentation [Текст] / R. Girshick [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2014. – С. 580–587.
62. Faster r-cnn: Towards real-time object detection with region proposal networks [Текст] / S. Ren [и др.] // Advances in neural information processing systems. – 2015. – Т. 28.
63. R-fcn: Object detection via region-based fully convolutional networks [Текст] / J. Dai [и др.] // Advances in neural information processing systems. – 2016. – Т. 29.
64. Libra r-cnn: Towards balanced learning for object detection [Текст] / J. Pang [и др.] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. – 2019. – С. 821–830.
65. Reppoints: Point set representation for object detection [Текст] / Z. Yang [и др.] // Proceedings of the IEEE/CVF international conference on computer vision. – 2019. – С. 9657–9666.
66. You only look once: Unified, real-time object detection [Текст] / J. Redmon [и др.] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – С. 779–788.
67. *Redmon J., Farhadi A.* YOLO9000: better, faster, stronger [Текст] // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2017. – С. 7263–7271.
68. Focal loss for dense object detection [Текст] / T.-Y. Lin [и др.] // Proceedings of the IEEE international conference on computer vision. – 2017. – С. 2980–2988.

69. Centernet: Keypoint triplets for object detection [Текст] / К. Duan [и др.] // Proceedings of the IEEE/CVF international conference on computer vision. – 2019. – С. 6569–6578.
70. *Law H., Deng J.* Cornernet: Detecting objects as paired keypoints [Текст] // Proceedings of the European conference on computer vision (ECCV). – 2018. – С. 734–750.
71. Cornernet-lite: Efficient keypoint based object detection [Текст] / Н. Law [и др.] // arXiv preprint arXiv:1904.08900. – 2019.
72. End-to-end object detection with transformers [Текст] / N. Carion [и др.] // European conference on computer vision. – Springer. 2020. – С. 213–229.
73. Spinenet: Learning scale-permuted backbone for recognition and localization [Текст] / X. Du [и др.] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. – 2020. – С. 11592–11601.
74. CSPNet: A new backbone that can enhance learning capability of CNN [Текст] / C.-Y. Wang [и др.] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. – 2020. – С. 390–391.
75. Spatial pyramid pooling in deep convolutional networks for visual recognition [Текст] / К. He [и др.] // IEEE transactions on pattern analysis and machine intelligence. – 2015. – Т. 37, № 9. – С. 1904–1916.
76. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs [Текст] / L.-C. Chen [и др.] // IEEE transactions on pattern analysis and machine intelligence. – 2017. – Т. 40, № 4. – С. 834–848.
77. Receptive field block net for accurate and fast object detection [Текст] / S. Liu, D. Huang [и др.] // Proceedings of the European conference on computer vision (ECCV). – 2018. – С. 385–400.

78. Cbam: Convolutional block attention module [Текст] / S. Woo [и др.] // Proceedings of the European conference on computer vision (ECCV). – 2018. – С. 3–19.
79. *Liu S., Huang D., Wang Y.* Learning spatial fusion for single-shot object detection [Текст] // arXiv preprint arXiv:1911.09516. – 2019.
80. M2det: A single-shot object detector based on multi-level feature pyramid network [Текст] / Q. Zhao [и др.] // Proceedings of the AAAI conference on artificial intelligence. Т. 33. – 2019. – С. 9259–9266.
81. *Белкина Г., Корсаков С.* По поводу выхода в свет 8-го издания «Философского словаря» под ред. ИТ Фролова [Текст] // Вопросы философии. – 2010. – № 3. – С. 177–182.
82. Improving neural networks by preventing co-adaptation of feature detectors [Текст] / G. E. Hinton [и др.] // arXiv preprint arXiv:1207.0580. – 2012.
83. Dropout: a simple way to prevent neural networks from overfitting [Текст] / N. Srivastava [и др.] // The journal of machine learning research. – 2014. – Т. 15, № 1. – С. 1929–1958.
84. *Ioffe S., Szegedy C.* Batch normalization: Accelerating deep network training by reducing internal covariate shift [Текст] // International conference on machine learning. – pmlr. 2015. – С. 448–456.
85. *Ковалев В. В.* Диагностика работы сверточных нейронных сетей методом GRAD-CAM [Текст] // Сборник трудов XVIII Всероссийской научной конференции молодых ученых, аспирантов и студентов «ИТСАУ-2020». – 2020. – С. 14–18.
86. Распознавание образов: состояние и перспективы [Текст] / К. Верхаген [и др.]. – 1985.
87. *Гонсалес Р., Вудс Р.* Цифровая обработка изображений [Текст]. – Litres, 2022.

88. Методы компьютерной обработки изображений [Текст] / М. В. Гашников [и др.]. – 2003.
89. *Воскобойников Ю. Е., Гочаков А. В., Колкер А. Б.* Фильтрация сигналов и изображений: Фурье и вейвлет алгоритмы [Текст]. – 2010.
90. *Фисенко В., Фисенко Т.* Компьютерная обработка и распознавание изображений [Текст]. – 2008.
91. *Коваль Ю. А., Филиппов М. В.* Метод предварительной фильтрации изображений для повышения точности распознавания образов [Текст] // Инженерный журнал: наука и инновации. – 2014. – 12 (36). – С. 12.
92. *Ковалев В. В., Сергеев Н. Е.* Алгоритм предварительной обработки видеоизображений для повышения точности обнаружения малоразмерных образов [Текст] // Известия Южного федерального университета. Технические науки. – 2021. – 5 (222). – С. 146–154.
93. *Ковалев В. В.* Алгоритм нелинейного контрастирования видеоизображений для повышения точности обнаружения объектов [Текст] // Материалы Всероссийской научно-технической конференции с международным участием имени профессора О.Н. Пьявченко «КомТех-2021». – 2021. – № 2. – С. 254–260.
94. *Рогачёв В., Ципуринда С.* Исследование методов комплексирования изображений различных диапазонов [Текст] // Труды учебных заведений связи. – 2016. – Т. 2, № 3. – С. 59–64.
95. *Krišto M., Ivašić-Kos M.* Thermal imaging dataset for person detection [Текст] // 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). – IEEE. 2019. – С. 1126–1131.
96. LLVIP: A visible-infrared paired dataset for low-light vision [Текст] / X. Jia [и др.] // Proceedings of the IEEE/CVF international conference on computer vision. – 2021. – С. 3496–3504.

97. Интеллектуальные информационные системы: Интеллектуальная информационная технология. Экспертные системы [Текст] / Д. Гаскаров [и др.] // СПб.: СПГУВК. – 2004.
98. Павлов С. Системы искусственного интеллекта [Текст]. – 2011.
99. Приказ Министерства экономического развития РФ от 29 июня 2021 г. № 392 «Об утверждении критериев определения принадлежности проектов к проектам в сфере искусственного интеллекта» [Электронный ресурс]. – URL: <https://minjust.consultant.ru/documents/27651?items=1&page=1> (дата обр. 23.07.2024).
100. Фреймворк глубокого обучения Tensorflow [Электронный ресурс]. – URL: <https://www.tensorflow.org/?hl=ru> (дата обр. 27.03.2024).
101. Фреймворк глубокого обучения Pytorch [Электронный ресурс]. – URL: <https://pytorch.org/> (дата обр. 27.03.2024).
102. Фреймворк глубокого обучения Keras [Электронный ресурс]. – URL: <https://keras.io> (дата обр. 27.03.2024).
103. Дж Ту Р. Принципы распознавания образов [Текст]. – Мир, 1978.
104. Патрик Э. А. Основы теории распознавания образов [Текст]. – 1980.
105. Шибанов Г. П. Распознавание в системах автоконтроля [Текст]. – 1973.
106. Wang C.-Y., Bochkovskiy A., Liao H.-Y. M. Scaled-yolov4: Scaling cross stage partial network [Текст] // Proceedings of the IEEE/cvf conference on computer vision and pattern recognition. – 2021. – С. 13029–13038.
107. Bochkovskiy A. Программные компоненты фреймворка ScaledYOLOv4 [Электронный ресурс]. – URL: <https://github.com/WongKinYiu/ScaledYOLOv4> (дата обр. 15.06.2024).
108. Misra D. Mish: A self regularized non-monotonic activation function [Текст] // arXiv preprint arXiv:1908.08681. – 2019.

109. *Redmon J.* Программные компоненты фреймворка Darknet [Электронный ресурс]. – URL: <https://github.com/pjreddie/darknet> (дата обр. 29.02.2024).
110. *Bochkovskiy A.* Программные компоненты фреймворка Darknet [Электронный ресурс]. – URL: <https://github.com/AlexeyAB/darknet> (дата обр. 29.02.2024).
111. *Ultralytics.* Фреймворк глубокого обучения YOLOv5 [Электронный ресурс]. – URL: <https://github.com/ultralytics/yolov5> (дата обр. 19.06.2024).
112. *ultralytics/yolov5: v6. 0-YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support* [Текст] / G. Jocher [и др.] // Zenodo. – 2021.
113. *Thuan D.* Evolution of Yolo algorithm and Yolov5: The State-of-the-Art object detection algorithm [Текст]. – 2021.
114. *Jocher G., Chaurasia A., Qiu J.* Ultralytics YOLOv8 [Текст]. – Вер. 8.0.0. – 2023. – URL: <https://github.com/ultralytics/ultralytics>.
115. *Terven J., Córdova-Esparza D.-M., Romero-González J.-A.* A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas [Текст] // Machine Learning and Knowledge Extraction. – 2023. – Т. 5, № 4. – С. 1680–1716.
116. Distance-IoU loss: Faster and better learning for bounding box regression [Текст] / Z. Zheng [и др.] // Proceedings of the AAAI conference on artificial intelligence. Т. 34. – 2020. – С. 12993–13000.
117. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection [Текст] / X. Li [и др.] // Advances in Neural Information Processing Systems. – 2020. – Т. 33. – С. 21002–21012.
118. Advances in neural information processing systems [Текст] / H. Lu [и др.] // Advances in neural information processing systems. – 2019. – Т. 32.

119. *Lucas B. D., Kanade T.* An iterative image registration technique with an application to stereo vision [Текст] // IJCAI'81: 7th international joint conference on Artificial intelligence. Т. 2. – 1981. – С. 674–679.
120. *Ковалев В. В.* Оптимизация гиперпараметров модели сверточной нейронной сети на основе предварительной обработки последовательности изображений [Текст] // Сборник статей VII Всероссийской научно-технической конференции «Фундаментальные и прикладные аспекты компьютерных технологий и информационной безопасности». – 2021. – С. 380–384.
121. *Ковалев В. В.* Разработка и исследование алгоритма повышения точности обнаружения малоразмерных объектов сверточными нейронными сетями [Текст] // Сборник статей IX Всероссийской научно-технической конференции «Фундаментальные и прикладные аспекты компьютерных технологий и информационной безопасности». – 2023. – С. 185–188.
122. *Ковалев В. В., Сергеев Н. Е.* Разработка, исследование алгоритма обнаружения объектов малых размеров на основе сверточных нейронных сетей [Текст] // Информатизация и связь. – 2023. – С. 18–22.
123. *Ковалев В. В.* Методы решения проблемы переобучения нейронных сетей в задаче обнаружения малоразмерных объектов на изображениях [Текст] // Труды международного научно-технического конгресса «Интеллектуальные системы и информационные технологии-2023». – 2023. – С. 38–45.
124. *Ковалев В. В., Сергеев Н. Е.* Расширение признакового пространства в задаче поиска и распознавания малоразмерных объектов на изображениях [Текст] // Известия ЮФУ. Технические науки. – 2024. – № 1.
125. Плагин инфракрасного диапазона для Unreal Engine 5 [Электронный ресурс]. – URL: <https://www.unrealengine.com/marketplace/en-US/product/0a289607a3f649a19e00184a852f267f> (дата обр. 19.06.2024).

126. *Витязев В., Колодько Г., Витязев С.* Способы и алгоритмы формирования радиолокационного изображения в режиме доплеровского облучения луча [Текст] // Цифровая обработка сигналов. – 2006. – № 3. – С. 31–41.
127. *Ковалев В. В., Сергеев Н. Е.* Реализация сверточных нейронных сетей на встраиваемых устройствах с ограниченным вычислительным ресурсом [Текст] // Известия Южного федерального университета. Технические науки. – 2021. – 6 (223). – С. 64–72.
128. *Ковалев В. В.* Алгоритм предварительной обработки изображений для снижения вероятности переобучения свёрточных нейронных сетей на нейронном ускорителе [Текст] // Известия Южного федерального университета. Технические науки. – 2024. – 5 (241). – С. 29–37.
129. *Ковалев В. В., Сергеев Н. Е.* Построение системы поиска и распознавания объектов на изображениях для устройств с ограниченным вычислительным ресурсом [Текст] // Сборник научных трудов XII Всероссийской научной конференции «Системный синтез и прикладная синергетика». – 2024. – С. 144–149.
130. *Подиновский В., Потапов М.* Метод взвешенной суммы критериев в анализе многокритериальных решений: pro et contra [Текст] // Бизнес-информатика. – 2013. – 3 (25). – С. 41–48.
131. Anchor box optimization for object detection [Текст] / Y. Zhong [и др.] // Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. – 2020. – С. 1286–1294.
132. *Hussain M., Bird J. J., Faria D. R.* A study on CNN transfer learning for image classification [Текст] // Advances in Computational Intelligence Systems: Contributions Presented at the 18th UK Workshop on Computational Intelligence, September 5-7, 2018, Nottingham, UK. – Springer. 2019. – С. 191–202.

133. Ускорение прямого прохода при реализации СНС на ограниченном вычислительном ресурсе [Текст] / А. Е. Щелкунов, В. В. Ковалев, И. В. Сидько, Н. Е. Сергеев // Известия Южного федерального университета. Технические науки. – 2022. – № 1. – С. 289–297.
134. *Vandersteegen M., Van Beeck K., Goedemé T.* Super accurate low latency object detection on a surveillance UAV [Текст] // 2019 16th International Conference on Machine Vision Applications (MVA). – IEEE. 2019. – С. 1–6.
135. YOLO nano: A highly compact you only look once convolutional neural network for object detection. arxiv 2019 [Текст] / A. Wong [и др.] // arXiv preprint arXiv:1910.01271. – 2019. – Т. 10.
136. *Stein E., Liu S., Sun J.* Real-time object detection on an edge device (Final Report) [Текст] // CS230: Deep Learning. – 2019.
137. Mixed precision training [Текст] / P. Micikevicius [и др.] // arXiv preprint arXiv:1710.03740. – 2017.
138. Bed: A real-time object detection system for edge devices [Текст] / G. Wang [и др.] // Proceedings of the 31st ACM International Conference on Information & Knowledge Management. – 2022. – С. 4994–4998.
139. *Горшенин А. К., Кузьмин В. Ю.* Оптимизация гиперпараметров нейронных сетей с использованием высокопроизводительных вычислений для предсказания осадков [Текст] // Информатика и её применения. – 2019. – Т. 13, № 1. – С. 75–81.

Приложение А

Листинги программ

Листинг А.1: Исходный код алгоритма предварительной обработки изображений с целью комплексирования признаков движения для центрального процессора на языке программирования *Python*

```
import cv2
import argparse
import numpy as np

5 class HistoryFrame:
    def __init__(self, indexes_queue: list):
        """
        Конструктор класса HistoryFrame
        :param indexes_queue: список индексов изображений в очереди
        """
        10 self.last_index = indexes_queue[-1]
        self.indexes_queue = [0] + indexes_queue
        self.queue = []

    15 def prepare_frame_uc8_to_float(self, frame: np.ndarray) -> np.ndarray:
        """
        Преобразование изображения во float32 с нормировкой
        :param frame: входное изображение в UC8
        :return: выходное изображение во float32
        """
        20 if len(frame.shape) >= 3:
            frame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
            frame = frame.astype(float)
            frame /= 255
        25 return frame

    def update(self, frame: np.ndarray) -> np.ndarray:
```

```

    """
    Обновление очереди изображений
30  :param frame: входное изображение в numpy
    :return: выходное изображение
    """
    frame = self.prepare_frame_uc8_to_float(frame)
    self.queue.append(frame)
35  if len(self.queue) > self.last_index:
        frame_fuse = cv2.merge([self.queue[self.last_index - i] for i in
self.indexes_queue])
        del self.queue[0]
        return frame_fuse
    else:
40  return cv2.merge([frame for _ in range(len(self.indexes_queue ))
])

def expansion_space(frame: np.ndarray):
    """
    Формирование комплексированного изображения
45  :param frame: входное изображение в формате float32 numpy
    :return: выходное комплексированное изображение
    """
    frame_fuse_diff = np.zeros_like(frame)
    frame_fuse_diff[..., 0] = frame[..., 0]
50  for i in range(1, frame.shape[2]):
        # формирование разностного изображения
        frame_diff = cv2.absdiff(frame[..., 0], frame[..., i])
        frame_fuse_diff[..., i] = frame_diff
    return frame_fuse_diff

```

Листинг А.2: Исходный код реализации алгоритма автоматического генерирования аннотированных условно-реальных данных изображений на языке программирования *Python*

```

import os
import cv2
import argparse
import numpy as np
5 from glob import glob
from typing import Union
from pathlib import Path

def save_data(image: np.ndarray, bboxes: list, dir_save: str, name: str,
10 suffix_im: str):
    """
    Пути сохранения аннотаций и изображений
    :param image: изображение в формате numpy
    :param bboxes: список аннотаций
    :param dir_save: выходная директория для сохранения данных
15 :param name: имя выходного файла
    :param suffix_im: суффикс изображения
    """
    path_save_image = os.path.join(dir_save, f"{name}.{suffix_im}")
    path_save_ann = os.path.join(dir_save, f"{name}.txt")
20 # сохранение изображения
    cv2.imwrite(path_save_image, image)
    # сохранение аннотаций
    with open(path_save_ann, "w") as f:
        for box in bboxes:
25             f.write(f"{box.get('id_class')} {box.get('center_x')} "
                    f"{box.get('center_y')} {box.get('box_x')} {box.get('
                    box_y')}\n")

class DataGenerator:
30     def __init__(self, dir_image: str, dir_mask: str, suffix_image: str,
                    path_mask_class: str, min_obj_sz: int):

```

```

"""
Конструктор класса DataGenerator
:param dir_image: директория с изображениями
:param dir_mask: директория с масками
35 :param suffix_image: суффикс изображения
:param path_mask_class: путь до файла с идентификаторами масок
объектов
:param min_obj_sz: минимальный размер объекта по Джонсону
"""
# формирование списка путей к изображениям и маскам
40 self.paths_images = glob(os.path.join(dir_image, f"*.{suffix_image}"
))
self.paths_masks = glob(os.path.join(dir_mask, f"*.{suffix_image}"))
self.min_obj_sz = min_obj_sz
# формирование словаря соответствий масок с классами
file = open(path_mask_class, "r")
45 lines = file.readlines()
self.mask_class = {int(line.split()[0]): int(line.split()[1]) for
line in lines}

def get_bboxes(self, image_mask: np.ndarray, width: int, height: int) →
list:
"""
50 Метод формирования списка аннотаций по изображению масок объектов
:param image_mask: изображение с масками объектов
:param width: ширина изображения
:param height: высота изображения
:return: список аннотаций
55 """
masks = []
# поиск значений масок объектов 8 бит ( нулевое значение фон)
for id_mask in range(1, 255):
y_mask, x_mask = np.where(image_mask == id_mask)
60 if (x_mask.size > 0) & (y_mask.size > 0):
# поиск класса по маске
id_class = self.mask_class.get(id_mask)
# определение обрамляющего прямоугольника (bbox)

```

```
        x_max, x_min, y_max, y_min = np.max(x_mask), np.min(x_mask),
np.max(y_mask), np.min(y_mask)
65         box_x = (x_max - x_min) / width
        box_y = (y_max - y_min) / height
        center_x = x_min / width + box_x / 2
        center_y = y_min / height + box_y / 2
        # условие малоразмерности
70         if ((x_max - x_min) | (y_max - y_min)) < self.min_obj_sz:
                continue
        masks.append(dict(id_class=id_class, center_x=center_x,
center_y=center_y, box_x=box_x, box_y=box_y))
        return masks

75     def __iter__(self) -> Union[np.ndarray, list]:
        """
        Метод генерации выходных данных
        :return: изображение в формате numpy,
                список аннотаций, имя файла
80         """
        for i, path_image in enumerate(self.paths_images):
            # чтение изображения
            image = cv2.imread(path_image, cv2.IMREAD_GRAYSCALE)
            width, height = image.shape[1], image.shape[0]
            # чтение маски
85            image_mask = cv2.imread(self.paths_masks[i], cv2.
IMREAD_GRAYSCALE)
            # определение bbox по маске объекта
            bboxes = self.get_bboxes(image_mask, width, height)
            yield image, bboxes, Path(path_image).stem
```

Листинг А.3: Исходный код алгоритма предварительной обработки изображений с целью комплексирования признаков движения для центрального процессора на языке программирования C++

```

#include <iostream>
#include <string>
#include <vector>
#include <opencv2/core.hpp>
5 #include <opencv2/videoio.hpp>
#include <opencv2/highgui.hpp>
#include <opencv2/imgproc/imgproc.hpp>
#include <chrono>

10

class ExpansionSpace{

public:

    std::vector<int> indexes_queue;
15    int last_index;
    std::vector<cv::Mat> queue;

    ExpansionSpace(std::vector<int> indexes_queue):
        last_index(indexes_queue[indexes_queue.size()-1]),
20        indexes_queue(indexes_queue)
    {}

    cv::Mat prepare_frame_uc8_to_float(cv::Mat frame){
        // Преобразование изображения во float32 с нормировкой
25

        cv::Mat dst_frame;
        frame.convertTo(dst_frame, CV_32F);
        return dst_frame / 255.0;

30    }

    cv::Mat update(cv::Mat frame){
        // Обновление очереди изображений

```

```
35 // и формирование комплексированного изображения

    if (frame.channels() == 3){
        cv::cvtColor(frame, frame, cv::COLOR_BGR2GRAY);
    }

40 std::vector<cv::Mat> frames_fuse_diff;
    cv::Mat result;

    frame = prepare_frame_uc8_to_float(frame);

45 queue.push_back(frame);

    if (queue.size() > last_index){
        frames_fuse_diff.push_back(queue.at(last_index));

50         for (int index: indexes_queue){
            cv::Mat diff;
            cv::absdiff(queue.at(last_index), queue.at(last_index -
index), diff);
            frames_fuse_diff.push_back(diff);
        }

55 queue.erase(queue.begin());
    }
    else{

        for (int i = 0; i < indexes_queue.size() + 1; i++){
60             frames_fuse_diff.push_back(frame);
        }
    }
    merge(frames_fuse_diff, result);
    return result;

65 }
```

Листинг А.4: Исходный код реализации алгоритма предварительной обработки изображений с целью комплексирования признаков движения для нейронного процессора на языке программирования *Python*

```

import cv2
import torch
import argparse
import numpy as np
5 import torch.nn as nn

class ExpansionSpaceCNN(nn.Module):

    def __init__(self, indexes_queue: list):
10         """
        Конструктор класса ExpansionSpaceCNN
        :param indexes_queue: список индексов изображений в очереди
        """
        super().__init__()
15         self.len_queue = indexes_queue[-1] + 1
        self.indexes_queue = [0] + indexes_queue
        self.queue = []
        self.in_channels = len(self.indexes_queue)
        self.out_channels = self.in_channels
20         self.conv_diff = nn.Conv2d(in_channels=self.in_channels,
                                     out_channels=self.out_channels,
                                     kernel_size=(1, 1),
                                     stride=(1, 1),
                                     bias=None)
25         self.weights_init()

    def weights_init(self):
30         """
        Инициализация весовых коэффициентов
        """
        weights_torch = torch.zeros((self.out_channels, self.in_channels, 1,
1))
        j = 0

```

```

    for i in range(self.indexes_queue[-1] + 1):
        if i in self.indexes_queue:
            weights_torch[j, ...][0] = 1
            if i > 0:
                weights_torch[j, ...][j] = -1
            j += 1
        self.conv_diff.weight.data = weights_torch

def forward(self, x: torch.Tensor) -> torch.Tensor:
    """
    Прямое распространение данные через модель
    :param x: входной тензор
    :return: выходной тензор
    """
    x = self.conv_diff(x)
    x = torch.abs(x)
    return x

def np_to_torch(np_frame: np.ndarray) -> torch.Tensor:
    """
    Преобразование данных numpy в torch формат
    :param np_frame: входное изображение в формате float32 numpy
    :return: изображение в формате torch.Tensor
    """
    torch_tensor = torch.Tensor(np_frame)
    torch_tensor = torch.unsqueeze(torch_tensor.permute(2, 0, 1), 0).cpu()
    return torch_tensor

def torch_to_np(tensor_fuse_diff: torch.Tensor) -> np.ndarray:
    """
    Преобразование данных torch в numpy формат
    :param tensor_fuse_diff: комплексированное изображение в формате torch.Tensor
    :return: комплексированное изображение в формате float32 numpy
    """
    tensor_fuse_diff = torch.squeeze(tensor_fuse_diff, 0).permute(1, 2, 0)
    frame_fuse_diff = tensor_fuse_diff.cpu().detach().numpy()

```

```

70     return frame_fuse_diff

class HistoryFrame:

75     def __init__(self, indexes_queue: list, imgsz: int):
        """
        Конструктор класса HistoryFrame
        :param indexes_queue: список индексов изображений
        :param imgsz: размер изображения
        """
80     self.last_index = indexes_queue[-1]
        self.indexes_queue = [0] + indexes_queue
        self.queue = []
        self.imgsz = imgsz

85     def prepare_frame_uc8_to_float(self, frame: np.ndarray) -> np.ndarray:
        """
        Преобразование изображения во float32 с нормировкой
        :param frame: входное изображение в UC8
        :return: выходное изображение во float32
90     """
        if len(frame.shape) >= 3:
            frame = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
            frame = frame.astype(float)
            frame /= 255
95     return frame

    def update(self, frame: np.ndarray) -> np.ndarray:
100     """
        Обновление последовательности кадров
        :param frame: входное изображение
        :return: выходное изображение
        """
        frame = cv2.resize(frame, (self.imgsz, self.imgsz))
        frame = self.prepare_frame_uc8_to_float(frame)
105     self.queue.append(frame)

```

```

        if len(self.queue) > self.last_index:
            frame_fuse = cv2.merge([self.queue[self.last_index - i] for i in
self.indexes_queue])
            del self.queue[0]
            return frame_fuse
110     else:
            return cv2.merge([frame for _ in range(len(self.indexes_queue ))
])

def export_onnx(model: nn.Module, name: str, imc: int, imw: int, im_h: int,
device: str):
    """
115     сохранение модели в ONNX формате
    :param model: pytorch модель
    :param name: путь сохранения ONNX
    :param imc: количество входных каналов изображения
    :param imw: ширина входного изображения
120     :param im_h: высота входного изображения
    :param device: тип нейроускорителя
    """
    torch.onnx.export(
125         model, torch.rand((1, imc, imw, im_h)).to(device),
        name,
        verbose=True, opset_version=11,
        operator_export_type=torch.onnx.OperatorExportTypes.
ONNX_ATEN_FALLBACK)

```

Приложение Б

Результаты вычислительного эксперимента построения интеллектуальной системы распознавания малоразмерных изображений на устройствах с ограниченным вычислительным ресурсом

Таблица 17 – Оценки значимости сформированных наборов

Выч. устройство	Архитектура СНС	Раз-ть вычислений	f_1 , Вт	f_2 , мс	f_3 , %	\hat{f}_1	\hat{f}_2	\hat{f}_3	$F(\hat{f} w)$
NVIDIA Jetson Nano B01	YOLOv3-Tiny-M	FP32	10	164	-	-	-	-	-
		FP16	10	110	-	-	-	-	-
	YOLOv5s-M	FP32	10	324	-	-	-	-	-
		FP16	10	196	-	-	-	-	-
	YOLOv5l-M	FP32	Недостаточный объём памяти						
		FP16	Недостаточный объём памяти						
	YOLOv8s-M	FP32	10	357	-	-	-	-	-
		FP16	10	220	-	-	-	-	-
NVIDIA Jetson TX2 (8GB)	YOLOv3-Tiny-M	FP32	15	75	-	-	-	-	-
		FP16	15	40	79.7	-	-	-	-
	YOLOv5s-M	FP32	15	124	-	-	-	-	-
		FP16	15	72	-	-	-	-	-
	YOLOv5l-M	FP32	15	512	-	-	-	-	-
		FP16	15	290	-	-	-	-	-
	YOLOv8s-M	FP32	15	140	-	-	-	-	-
		FP16	15	81	-	-	-	-	-
NVIDIA Jetson AGX Xavier	YOLOv3-Tiny-M	FP32	30	27	79.6	-	-	-	-
		FP16	30	10	79.7	-	-	-	-
		INT8	30	8	63.1	-	-	-	-
	YOLOv5s-M	FP32	30	51	86.5	1	0.43	0.92	0.76
		FP16	30	22	86.4	1	1	0.92	0.98
		INT8	30	13	73.6	-	-	-	-
	YOLOv5l-M	FP32	30	191	-	-	-	-	-
		FP16	30	58	93.4	1	0.37	1	0.75
		INT8	30	32	87.8	1	0.68	0.93	0.86
	YOLOv8s-M	FP32	30	47	83.8	-	-	-	-
		FP16	30	17	83.7	-	-	-	-
		INT8	30	12	78.4	-	-	-	-
Rockchip RK3588	YOLOv3-Tiny-M	INT8	12	30	65.2	-	-	-	-
	YOLOv5s-M	INT8	12	60	76.1	-	-	-	-
	YOLOv5l-M	INT8	12	230	-	-	-	-	
	YOLOv8s-M	INT8	12	70	-	-	-	-	

Приложение В

Акты о внедрении результатов работы

Копии актов о внедрении результатов диссертационной работы приведены на следующих страницах.

УТВЕРЖДАЮ



Генеральный директор
Акционерное общество
«Научно-конструкторское бюро
вычислительных систем»

О.В. Максёнков

01 _____ 2025 г.

АКТ

о внедрении в научно-исследовательской работе «Гепард-Деталь-НКБ ВС-БЦВМ-ЭИИ» результатов, полученных в диссертации Ковалева Владислава Владимировича на тему «МЕТОД И АЛГОРИТМЫ РАСПОЗНАВАНИЯ МАЛОРАЗМЕРНЫХ ИЗОБРАЖЕНИЙ ПОДВИЖНЫХ ОБЪЕКТОВ НА УСТРОЙСТВАХ С ОГРАНИЧЕННЫМ ВЫЧИСЛИТЕЛЬНЫМ РЕСУРСОМ», представленной на соискание учёной степени кандидата технических наук

Настоящий Акт свидетельствует о том, что теоретические и практические результаты диссертации Ковалева Владислава Владимировича были внедрены в научно-исследовательской работе «Гепард-Деталь-НКБ ВС-БЦВМ-ЭИИ», выполненной в АО Научно-конструкторское бюро вычислительных систем в 2021 году:

1. Алгоритм предварительной обработки изображения с целью комплексирования признаков движения на центральном процессоре, который позволяет повысить качество распознавания малоразмерных изображений свёрточными нейронными сетями;
2. Алгоритм формирования условно-реальных данных, который позволяет генерировать в автоматическом режиме аннотированные условно реальные разномодальные изображения.

Зав. отделом 51

АО Научно-конструкторское бюро
вычислительных систем, к.т.н.

И.В. Сидько

УТВЕРЖДАЮ

Проректор по образовательной
деятельности и информационным
технологиям ФГАОУ Южный
Федеральный университет
кандидат физико-математических
наук, доцент




Н.В. Махно

« 8 » февраля 2025 г.

АКТ

о использовании в учебном процессе кафедры вычислительной техники результатов, полученных в диссертации Ковалева Владислава Владимировича на тему «МЕТОД И АЛГОРИТМЫ РАСПОЗНАВАНИЯ МАЛОРАЗМЕРНЫХ ИЗОБРАЖЕНИЙ ПОДВИЖНЫХ ОБЪЕКТОВ НА УСТРОЙСТВАХ С ОГРАНИЧЕННЫМ ВЫЧИСЛИТЕЛЬНЫМ РЕСУРСОМ», представленной на соискание учёной степени кандидата технических наук

Комиссия в составе: А.Е. Лызь к.т.н., зам. директора Института компьютерных технологий и информационной безопасности (ИКТИБ) ЮФУ, А.Н. Самойлова – к.т.н., заведующего кафедрой вычислительной техники (ВТ), В.Н. Пуховского – к.т.н., доцента кафедры ВТ, руководителя основной профессиональной образовательной программы бакалавриата «Технологии искусственного интеллекта» по направлению 09.03.01 Информатика и вычислительная техника, М.Ю. Поленова – к.т.н., доцента кафедры ВТ, руководителя основной профессиональной образовательной программы магистратуры «Высокопроизводительные вычислительные системы и квантовая обработка информации» по направлению 09.04.01 Информатика и вычислительная техника, настоящим актом подтверждает, что следующие теоретические и практические результаты диссертации Ковалева Владислава Владимировича используются в учебном процессе для формирования компетенций обучающихся:

1. Метод расширения исходного признакового пространства, который позволяет повысить информативность системы признаков за счёт добавления признаков движения;
2. Алгоритм предварительной обработки изображения с целью комплексирования признаков движения на центральном процессоре, который позволяет повысить качество распознавания малоразмерных изображений свёрточными нейронными сетями;
3. Алгоритм формирования условно-реальных данных, который позволяет генерировать в автоматическом режиме аннотированные условно-реальные разномодальные изображения;
4. Алгоритм предварительной обработки изображений с целью комплексирования признаков движения на нейронном ускорителе, который позволяет повысить скорость обработки данных;
5. Алгоритм построения интеллектуальной системы распознавания малоразмерных изображений объектов на устройствах с ограниченным вычислительным ресурсом, которая удовлетворяет требуемым характеристикам.

Перечисленные результаты используются в учебном процессе при проведении занятий по дисциплинам:

1. Представление и использование знаний в интеллектуальных системах;
2. Высокопроизводительные вычислительные системы.

Зам. директора

Института компьютерных технологий и
информационной безопасности, к.т.н.

А.Е. Лызь

Заведующий

кафедры вычислительной техники, к.т.н.

А.Н. Самойлов

Доцент

кафедры вычислительной техники, к.т.н.

В.Н. Пуховский

Доцент

кафедры вычислительной техники, к.т.н.

М.Ю. Поленов