

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт кибернетики
Направление подготовки 09.04.02 «Информационные системы и технологии»
Кафедра информационных систем и технологии

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Разработка и программная реализация алгоритма детектирования номерных знаков на изображениях

УДК 004.415:629.3.066.8:004.932.75'1

Студент

Группа	ФИО	Подпись	Дата
8ИМ5А	Сыздыкова Г.Ж.		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. ИСТ	Болотова Ю.А.	к.т.н		

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Попова С.Н.	к.э.н		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент каф. ЭБЖ	Акулов П.А			

ДОПУСТИТЬ К ЗАЩИТЕ:

Зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
ИСТ	Мальчуков А.Н.	к.т.н		

Томск – 2017 г.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Код результатов	Результат обучения (выпускник должен быть готов)
Общепрофессиональные компетенции	
P1	Воспринимать и самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте.
P2	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях.
P3	Демонстрировать культуру мышления, способность выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных, анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.
P4	Анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего образования и профессиональной мобильности. Владеть, по крайней мере, одним из иностранных языков на уровне социального и профессионального общения, применять специальную лексику и профессиональную терминологию языка.
Профессиональные компетенции	
P5	Разрабатывать стратегии и цели проектирования, критерии эффективности и ограничения применимости, новые методы, средства и технологии проектирования геоинформационных систем (ГИС) или промышленного программного обеспечения.
P6	Планировать и проводить теоретические и экспериментальные исследования в области создания интеллектуальных ГИС и ГИС технологии или промышленного программного обеспечения с использованием методов системной инженерии.
P7	Осуществлять авторское сопровождение процессов проектирования, внедрения и сопровождения ГИС и ГИС технологий или промышленного программного обеспечения с использованием методов и средств системной инженерии, осуществлять подготовку и обучение персонала.

P8	Формировать новые конкурентоспособные идеи в области теории и практики ГИС и ГИС технологий или системной инженерии программного обеспечения. Разрабатывать методы решения нестандартных задач и новые методы решения традиционных задач. Организовывать взаимодействие коллективов, принимать управленческие решения, находить компромисс между различными требованиями как при долгосрочном, так и при краткосрочным планировании.
Общекультурные компетенции	
P9	Использовать на практике умения и навыки в организации исследовательских, проектных работ и профессиональной эксплуатации современного оборудования и приборов, в управлении коллективом.
P10	Свободно пользоваться русским и иностранным языками как средством делового общения.
P11	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.
P12	Демонстрировать способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности, способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, способность к педагогической деятельности.

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт кибернетики

Направление подготовки (специальность) 09.04.02 «Информационные системы и технологии»

Кафедра Информационных систем и технологии

УТВЕРЖДАЮ:

Зав. кафедрой

_____ Мальчуков А.Н.
(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

В форме:

магистерской диссертации

Студенту:

Группа	ФИО
8ИМ5А	Сыздыковой Гаухар Жумагулкызы

Тема работы:

Разработка и программная реализация алгоритма детектирования номерных знаков на изображениях	
Утверждена приказом директора (дата, номер)	№ 897/с от 20.02.2017

Срок сдачи студентом выполненной работы:

--	--

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	Разработка и программная реализация алгоритма детектирования номерных знаков на изображениях. Алгоритм детектирования номерных знаков может применяться в правоохранительных органах, в системах контроля автомобилей и транспортных средств.
---------------------------------	---

Перечень подлежащих исследованию, проектированию и разработке вопросов	Обзор и анализ основных методов детектирования автомобильных номерных знаков, проектирование структуры и содержания основных классов инфраструктуры разрабатываемого программного обеспечения, разработка основных компонентов программного обеспечения, расчет ресурсоэффективности и ресурсосбережения, анализ вредных производственных факторов.
Перечень графического материала	Блок-схема алгоритма, структура основных классов библиотеки, изображения результатов фильтрации и выравнивания гистограммы, изображения работы алгоритма детектирования номерных знаков.

Консультанты по разделам выпускной квалификационной работы

Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Попова С.Н.
Социальная ответственность	Акулов П.А.
Раздел на иностранном языке	Горбатова Т.Н.

Названия разделов, которые должны быть написаны на русском и иностранном языках:

Разделы введение, аналитический обзор, алгоритм и программные средства детектирования номерных знаков, результаты тестирования, финансовый менеджмент, ресурсоэффективность и ресурсосбережение, социальная ответственность и заключение должны быть написаны на русском языке.

Раздел аналитический обзор должен быть написан на английском языке.

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
---	--

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. ИСТ	Болотова Ю.А.	к.т.н		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5А	Сыздыкова Г.Ж.		

Министерство образования и науки Российской Федерации
 федеральное государственное автономное образовательное учреждение
 высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
 ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт Кибернетики

Направление подготовки (специальность) 09.04.02 «Информационные системы и технологии»

Уровень образования: магистр

Кафедра Информационных систем и технологий

Период выполнения: осенний семестр 2016 г. – весенний семестр 2017 учебного года

Форма представления работы:

Магистерская диссертация

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
 выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:	
--	--

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
27.06.2016	<i>Постановка задачи и анализ предметной области</i>	10
28.08.2016	<i>Подготовка обучающей и тестовой выборки</i>	5
12.09.2016	<i>Реализация метода контурного анализа</i>	15
12.09.2016	<i>Реализация метода Виолы - Джонса</i>	15
12.09.2016	<i>Сопоставление методов детектирования номерных знаков</i>	10
10.11.2016	<i>Составление плана для модификации метода Виолы – Джонса</i>	5
14.02.2017	<i>Реализация методов сглаживания изображения и выравнивания гистограммы</i>	10
14.02.2017	<i>Разработка статистического метода описания текстурных областей изображения</i>	15
15.05.2017	<i>Тестирование и анализ результатов модифицированного метода Виолы – Джонса</i>	10
01.06.2017	<i>Оформление пояснительной записки</i>	5

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. ИСТ	Болотова Ю.А.	к.т.н		

СОГЛАСОВАНО:

Зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
ИСТ	Мальчуков А.Н.	к.т.н		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСООБЪЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
8ИМ5А	Сыздыковой Гаухар Жумагулкызы

Институт	Институт кибернетики	Кафедра	Информационных систем и технологии
Уровень образования	Магистр	Направление/специальность	09.04.02 Информационных систем и технологии

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

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	<i>На основании информации, представленной в научных статьях и публикациях, аналитических материалах, статистических бюллетенях и изданиях, нормативно-правовых документах, определить методику расчета экономической эффективности.</i>
2. Нормы и нормативы расходования ресурсов	
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Оценка коммерческого потенциала инженерных решений (ИР)	<i>Оценка ресурсной, социальной эффективности НИ и потенциальных рисков.</i>
2. Формирование плана и графика разработки и внедрения ИР	<i>Планирование этапов работы, определение календарного графика и трудоемкости разработки.</i>
3. Составление бюджета инженерного проекта (ИП)	<i>Затраты на материальные ресурсы, электроэнергию, заработную плату, страховые взносы, накладные расходы.</i>
4. Оценка ресурсной, финансовой, социальной, бюджетной эффективности исследования	<i>Оценка сравнительной эффективности проекта</i>

Перечень графического материала (с точным указанием обязательных чертежей)

1. Перечень работ и продолжительность их выполнения
2. Трудозатраты на выполнение проекта
3. Линейный график работ
4. Нарастание технической готовности работы и удельный вес каждого этапа
5. Расчет затрат на материалы
6. Затраты на заработную плату
7. Затраты на электроэнергию технологическую
8. Смета затрат на разработку проекта
9. Оценки научно-технического уровня НИР

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

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Попова С.Н.	к.э.н		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5А	Сыздыкова Гаухар Жумагулкызы		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8ИМ5А	Сыздыковой Гаухар Жумагулкызы

Институт	Институт киберне- тики	Кафедра	Информационных систем и технологии
Уровень образования	Магистр	Направление/специаль- ность	09.04.02 Информационных систем и технологии

Исходные данные к разделу «Социальная ответственность»:

<i>1. Характеристика объекта исследования и области его применения</i>	<i>Целью магистерской диссертации является разработка и программная реализация алгоритма детектирования номерных знаков на изображениях. Алгоритм детектирования номерных знаков может применяться в правоохранительных органах, в системах контроля автомобилей и транспортных средств.</i>
--	--

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Профессиональная и социальная безопасность <i>1.1. Анализ выделенных вредных и опасных факторов, которые могут возникнуть при разработке проектируемого решения.</i> <i>1.2. Рекомендации по защите исследователя от действия опасных и вредных факторов.</i>	<i>1.1. В качестве вредных факторов выделены:</i> –Нарушение параметров микроклимата; –Недостаточная освещенность рабочей зоны; –Монотонный режим работы. – Опасность поражения электрическим током; – Опасность возникновения пожара. <i>1.2. Рекомендации по защите исследователя от действия опасных и вредных факторов.</i>
2. Экологическая безопасность <i>2.1. Анализ влияния процесса разработки на окружающую среду.</i> <i>2.2. Обоснование мероприятий по защите окружающей среды.</i>	<i>2.1. Деятельность организации не связана с производством, следовательно, влияние на окружающую среду минимально.</i> <i>2.2. Рассмотрена утилизация бумажных отходов и компьютерной техники, приведены дальнейшие сферы применения проектируемого решения.</i>
3. Безопасность в чрезвычайных ситуациях <i>3.1. Перечень возможных ЧС, которые могут возникнуть при разработке проектируемого решения.</i> <i>3.2. Меры по ликвидации ЧС и последствий</i>	<i>3.1 Вероятные ЧС:</i> – Возникновение пожара в помещении; – Кибертерроризм. <i>3.2 Мероприятия по предотвращению наиболее типичной ЧС – пожара, согласно нормативным документам:</i> – НПБ 105-03; – ППБ 01–03.
4. Правовые и организационные вопросы обеспечения безопасности <i>4.1 Специальные (характерные для про-</i>	<i>4.1 Описание правовых норм для работ, связанных с работой за ПЭВМ согласно следующим документам:</i>

<p>ектируемой рабочей зоны) правовые нормы трудового законодательства. 4.2. Организационные мероприятия при компоновке рабочей зоны</p>	<p>– Трудовой кодекс Российской Федерации" от 30.12.2001 N 197-ФЗ (ред. от 30.12.2015). 4.2 Влияние реализованного алгоритма детектирования номерных знаков на работу систем контроля автомобилей и транспортных средств: – Экономия времени и денежных средств. – Преимущества автоматического детектирования над визуальным контролем.</p>
---	--

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

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Акулов Петр Анатольевич			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5А	Сыздыкова Гаухар Жумагулқызы		

РЕФЕРАТ

Выпускная квалификационная работа содержит 126 с., 27 рис., 19 табл., 55 источников, 3 прил. (включая CD - диск).

Ключевые слова: детектирование автомобильных номеров, контурный анализ, метод Виолы – Джонса, предобработка изображения, текстурные особенности, цифровая обработка изображений.

Объектом исследования является задача определения расположения номерной пластины на изображении.

Цель работы – разработка и программная реализация алгоритмов детектирования номерных знаков.

В процессе исследования были изучены и проанализированы существующие методы детектирования номерных знаков, методы текстурной сегментации изображений и методы предобработки изображения.

В результате исследования был предложен и реализован алгоритм детектирования номерных знаков на изображениях.

Область применения: в системах контроля и регистрации автомобилей и других видов транспортных средств.

В будущем планируется развитие работы в направлении улучшения и расширения реализованного алгоритма.

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

OpenCV – библиотека компьютерного зрения с открытым исходным кодом;

ВКР – выпускная квалификационная работа;

НР – научный руководитель;

И – исполнитель;

НТИ – научно-техническое исследование;

НДС – налог на добавленную стоимость;

ПО – программное обеспечение;

ПК – персональный компьютер;

ПЭВМ – персональная электронно – вычислительная машина;

ЧС – чрезвычайная ситуация;

СанПиН – санитарные нормы и правила;

РФ – Российская Федерация;

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	15
1 АНАЛИТИЧЕСКИЙ ОБЗОР	17
1.1 Метод контурного анализа	17
1.1.1 Детектор границ Канни	19
1.1.2 Алгоритм Дугласа-Пекера для аппроксимации контуров	21
1.2 Метод Виолы—Джонса	22
1.2.1 Интегральное изображение.....	22
1.2.2 Признаки Хаара	24
1.2.3 Классификатор на основе алгоритма адаптивного бустинга	25
1.2.4 Схема обучения	27
1.3 Методы сглаживания изображения.....	29
1.3.1 Линейные фильтры	30
1.3.2 Фильтр Гаусса	31
1.4 Выравнивание гистограммы.....	35
1.5 Методы описания текстурных областей изображения.....	38
1.6. Вывод по Главе 1	41
2 АЛГОРИТМ И ПРОГРАММНЫЕ СРЕДСТВА	42
2.1. Разрабатываемый алгоритм детектирования номерных знаков	42
2.2 Разработка программного обеспечения	50
2.2.1 Язык программирования C++	50
2.2.2 Библиотека OpenCV	51
2.3 Описание разработанного ПО	53
2.4 Выводы по Главе 2.....	62
3 РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ	63
3.1 Сравнение метода контурного анализа и метода Виолы - Джонса	64
3.2 Тестирование предложенного метода детектирования.....	68
3.2.1 Вычисление статистических характеристик текстур	72
3.3 Результаты экспериментальной части	73
3.4 Сравнение алгоритмов.....	74

4	ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ	77
4.1	Организация и планирование работы.....	77
4.2	Продолжительность этапов работ	78
4.3	Расчет сметы затрат на выполнение проекта.....	81
4.3.1	Расчет заработной платы	81
4.3.2	Расчет затрат на социальный налог	82
4.3.3	Расчет затрат на электроэнергию	82
4.3.4	Расчет прочих расходов	83
4.3.5	Расчет общей себестоимости разработки.....	83
4.3.6	Расчет прибыли	84
4.4	Оценка экономической эффективности проекта	84
4.5	Оценка научно-технического уровня НИР	85
4.6	Выводы по Главе 4	86
5	СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ	87
5.1	Профессиональная и социальная безопасность.....	87
5.1.1	Анализ выделенных вредных и опасных факторов, которые могут возникнуть при разработке проектируемого решения.....	87
5.1.2	Рекомендации по защите исследователя от действия опасных и вредных факторов	94
5.2	Экологическая безопасность	95
5.2.1	Анализ влияния процесса разработки на окружающую среду	95
5.2.2	Обоснование мероприятий по защите окружающей среды.....	96
5.3	Безопасность в чрезвычайных ситуациях	97
5.3.1	Перечень возможных ЧС, которые могут возникнуть при разработке.....	97
5.3.2	Меры по ликвидации ЧС и последствий.....	98
5.4	Правовые и организационные вопросы обеспечения безопасности	100
5.4.1	Специальные правовые нормы трудового законодательства	100
5.4.2	Организационные мероприятия при компоновке рабочей зоны	101
	ЗАКЛЮЧЕНИЕ	102

ОПУБЛИКОВАННЫЕ РАБОТЫ	103
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	104
Приложение А	109
Приложение Б.....	126

ВВЕДЕНИЕ

Компьютерное зрение – область знаний, изучающая создание машинных алгоритмов, которые могут производить обнаружение, отслеживание и классификацию объектов на изображениях [1]. Цель компьютерного зрения – научить компьютер распознавать объекты на статическом изображении или видеозаписях. При решении задач компьютерного зрения на вход подаётся некоторое изображение, а на выходе возможно получение признаков всех важных (в зависимости от конкретных реализаций и предметных областей) объектов и их классификации.

Поиск и локализация объектов на изображении – важное направление в компьютерном зрении. Одна из первых работ по данной теме была написана ещё в 1960 году [1]. Несмотря на то, что в настоящее время существует большое множество алгоритмов (которые на практике зачастую комбинируются между собой), новые алгоритмы продолжают создаваться и настоящее время, что говорит об актуальности данной проблемы.

Реализация систем компьютерного зрения сильно зависит от области их применения, аппаратной платформы и требований производительности. К новейшим областям применения относятся автономные транспортные средства.

Цель данной работы – разработка и реализация алгоритма детектирования номерных знаков.

Данная работа предполагает решение следующих задач:

- 1) исследование, реализация и сопоставление двух методов детектирования объектов на изображениях.
- 2) исследование и применение методов предобработки изображений, таких как фильтры сглаживания, эквализация гистограммы и статистические показатели текстурных особенностей изображений.
- 3) Разработка собственного алгоритма детектирования номерных знаков автомобилей.
- 4) Проведение численных экспериментов.

В первом разделе описана основная теоритическая часть: исследуемые методы детектирования номерных знаков, методы предобработки изображения и методы описания текстурных областей изображения.

Во втором разделе описан предложенный алгоритм и реализованное в ходе работы программное обеспечение для детектирования номерных знаков. В результате сопоставления различных вариантов предложенного алгоритма, был выбран вариант, показывающий наилучшую точность детектирования номерной пластины при наименьшем количестве ложных срабатываний.

В третьем разделе приведены результаты тестирования разработанного ПО для детектирования номерных знаков.

В четвертом и пятом разделах приведены материалы, связанные с ресурсоэффективностью и социальной ответственностью.

Научная новизна работы заключается в разработанном алгоритме детектирования номерного знака на изображениях..

По теме диссертации были сделаны две публикации в международных научно-практических конференциях студентов, аспирантов и молодых ученых:

1. XIV Международная научно-практическая конференция студентов, аспирантов и молодых ученых «Молодежь и современные информационные технологии» 7 – 11 ноября, 2016 г., г. Томск.
2. Международная научно-техническая конференция студентов, аспирантов и молодых учёных «Научная сессия ТУСУР – 2017», посвящённая 55-летию ТУСУРа, 10 по 12 мая, 2017.

1 АНАЛИТИЧЕСКИЙ ОБЗОР

Локализация номерного знака на изображении является чрезвычайно важной, так как от того, насколько аккуратно выделен номерной знак на изображении, зависят результаты всех последующих стадий: сегментации и распознавания. В настоящее время для выделения номерных знаков может быть применен целый ряд методов: разбиение изображения на блоки; сравнение с шаблоном (область, имеющая наибольшее сходство с шаблоном, сегментируется как искомый номерной знак); морфологические операторы; преобразование Хафа, с помощью которого можно находить различные формы в изображении (круг, эллипс, прямая); нахождение границ изображения с использованием таких алгоритмов как алгоритм Собеля [2], Канни, Робинсона [3] и др; метод Виолы-Джонса [4]; методы, основанные на анализе гистограммы изображения [5] и выявлении наиболее контрастных областей. Одними из самых эффективных подходов для решения задачи локализации объектов на изображении являются метод контурного анализа [6] и метод Виолы-Джонса [4]. С помощью данных методов во множестве работ получены очень хорошие результаты обнаружения различных сложных объектов, они имеют высокую производительность [3].

1.1 Метод контурного анализа

Контурный анализ – это один из важных и очень полезных методов описания, хранения, распознавания, сравнения и поиска графических образов или объектов.

Границы являются кривыми на изображении, вдоль которых происходят резкие перепады яркости [7]. Контур является внешним очертанием объекта. В контурном анализе контур описывает и содержит необходимую информацию о форме объекта. В таком подходе внутренние точки объекта не учитываются. Это позволяет сократить время обработки информации.

При выделении контуров изображения может возникнуть ряд следующих проблем:

- наличие разрывов контура в областях изображения, где яркость меняется не слишком быстро;
- наличие ложных контуров из-за шума на изображении;
- наличие широких контурных линий из-за размытости или шума.

С помощью контурного анализа можно описывать, хранить, сравнивать и обнаруживать объекты, представленные в виде своих внешних очертаний – контуров, а также эффективно решать такие задачи, как перемещение, поворот и изменение масштаба изображения объекта.

При проведении контурного анализа, контур кодируется последовательностью. Последовательность состоит из комплексных чисел. Поначалу на контуре отмечается начальная точка. После этого контур обводится, например, по направлению движения часовой стрелки. Каждый вектор перемещения записывается комплексным числом $a + ib$, где a – перемещение точки по оси x ; b – перемещение точки по оси y . Перемещение берется относительно предыдущей точки (на рисунке 1.1) [8].



Рисунок 1.1 – Кодирование контура

Последовательность действий при распознавании объектов методом контурного анализа выглядит следующим образом:

1. Предварительная обработка изображения.
2. Выделение границ объектов.

3. Нахождение контуров на изображении.
4. Работа с параметрами контура.

1.1.1 Детектор границ Канни

Детектор границ Канни является самым популярным методом выделения границ. Был разработан в 1986 году Джоном Канни (англ. John F. Canny). В детекторе используется многоступенчатый алгоритм для обнаружения огромного множества границ в изображениях.

Основные этапы алгоритма:

1. Сглаживание. Размытие изображения для удаления шума. В операторе Канни используется фильтр, который может быть хорошо приближенным к первой производной гауссианы, $\delta = 1.4$:

$$B = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} * A. \quad (1.1)$$

2. Поиск градиентов происходит таким образом: границы фиксируются в местах, где градиент изображения достигает максимального значения. Направления градиентов могут быть различными. Поэтому для обнаружения горизонтальных, вертикальных и диагональных ребер в размытом изображении в алгоритме Канни применяется четыре фильтра. Определение значения первой производной в горизонтальном (G_y) и вертикальном направлениях (G_x) происходит с использованием операторов обнаружения границ, например, с помощью оператора Собеля [9].

$$G = \sqrt{G_x^2 + G_y^2}. \quad (1.2)$$

На основе информации о градиенте производится расчет угла направления границы:

$$\theta = \arctg\left(\frac{G_y}{G_x}\right). \quad (1.3)$$

Определяются углы, отображающие вертикаль, горизонталь и две диагонали. Округление угла направления границы производится до одной из этих четырех углов.

3. При подавлении немаксимумов идет проверка величины градиента, достигает ли он локального максимума в соответствующем направлении.

В качестве кандидатов на группировку в край выбираются точки, достигшие локального максимума.

Можно привести пример для окна размером 3×3 . Для наглядности угол направления градиента обозначим как f , а интенсивность I .

1) точка будет приниматься границей, если её значение I больше чем у точки расположенной выше и ниже рассматриваемой точки, при условии, что $f=0^\circ$,

2) точка будет приниматься границей, если её значение I больше чем у точки расположенной слева и справа рассматриваемой точки, при условии, что $f = 90^\circ$.

3) точка будет приниматься границей, если её значение I больше чем у точек расположенных в верхнем левом и нижнем правом углу от рассматриваемой точки, при условии, что $f = 135^\circ$.

4) точка будет приниматься границей, если её значение I больше чем у точки находящейся в верхнем правом углу и больше чем у точки находящейся в нижнем левом углу от рассматриваемой точки, при условии, что $f = 45^\circ$ [9].

В результате получается изображение, содержащее границы объектов.

4. Двойной порог: потенциальные ребра определяются путем установления порога.

5. Итоговые границы: края, не связанные с сильными границами, подавляются и устанавливаются конечные границы.

1.1.2 Алгоритм Дугласа-Пекера для аппроксимации контуров

Алгоритм Дугласа-Пекера — это алгоритм, позволяющий уменьшить число точек кривой, аппроксимированной большей серией точек. Алгоритм был независимо открыт Урсом Рамером в 1972 и Давидом Дугласом и Томасом Пекером в 1973 [10].

Входом алгоритму Дугласа-Пекера служат аппроксимируемый контур и заданное расстояние $\varepsilon > 0$. ε — это наиболее удаленное расстояние между точками контура. Было установлено, что для достижения наилучшего результата аппроксимации ε имеет смысл брать равным $0,02 \times L$, где L — длина аппроксимируемого контура [10].

Алгоритм рекурсивно делит линию. Приведем пошаговое описание алгоритма:

Инициализация — берется прямая, идущая от начальной точки к конечной. Если контур замкнутый — то выбираются точки, максимально удаленные друг от друга. На рисунках 1.2 — 1.5 приведены результаты пошаговой работы алгоритма.

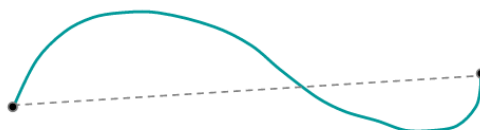


Рисунок 1.2 – Ход работы алгоритма Дугласа-Пекера

Шаг 1 — найти точку контура максимально удаленную от прямой. Если расстояние от нее до прямой $d < \varepsilon$ — разбиение завершено, если нет — к шагу 2.

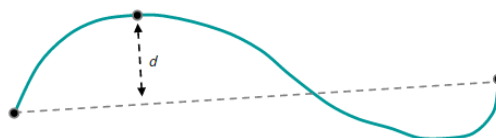


Рисунок 1.3 – Первая итерация работы алгоритма Дугласа-Пекера

Шаг 2 — добавить найденный на предыдущем шаге узел к ломаной линии. Затем рекурсивно вызвать Шаг 1 для каждого из половинок ломаной.

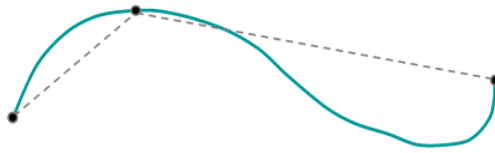


Рисунок 1.4 – Вторая итерация работы алгоритма Дугласа-Пекера

Результат третьей итерации работы алгоритма приведена на рис. 1.5:

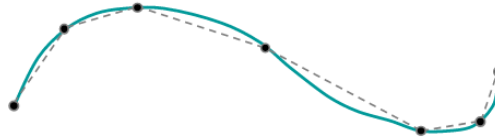


Рисунок 1.5 – Третья итерация работы алгоритма Дугласа-Пекера

Выходная ломаная строится по завершению всех рекурсивных вызовов, из точек, отмеченных к сохранению [10].

1.2 Метод Виолы—Джонса

Метод Виолы – Джонса (англ. Viola–Jones object detection) – алгоритм, обнаружения объектов на изображениях в реальном времени, предложен в 2001 году П. Виолой и М. Джонсом [11].

В настоящее время метод Виолы–Джонса является распространенным методом для обнаружения объекта на изображении благодаря своей высокой скорости и эффективности. Основами метода Виолы–Джонса выступают: интегральное представление изображения, признаки Хаара, построение классификатора на основе алгоритма бустинга и метод объединения классификаторов в каскадную структуру. Эти принципы дают возможность осуществлять поиск объекта в режиме реального времени.

1.2.1 Интегральное изображение

Интегральное представление изображений используется для осуществления определенных действий с данными. Интегральное представление изображения представляет собой матрицу, одинакового размера

с исходным изображением. В каждом элементе матрицы хранится суммарная яркость всех пикселей, расположенных левее и выше текущего элемента – правого нижнего угла прямоугольной области $(0, 0)$ до (x, y) . По следующей формуле можно рассчитать элементы матрицы L [4]:

$$(L(x, y) = \sum_{i=0, j=0}^{i \leq x, j \leq y} I(i, j), \quad (1.4)$$

где $I(i, j)$ – яркость пиксела исходного изображения.

Расчёт значений элементов матрицы проходит за время, пропорциональное количеству пикселей в текущем изображении. Таким образом интегральное изображение считывается за один проход.

Подсчет элементов матрицы производится по формуле:

$$L(x, y) = I(x, y) - L(x - 1, y - 1) + L(x, y - 1) + L(x - 1, y), \quad (1.5)$$

После выстаривания подобной интегральной матрицы сумму пикселей произвольной прямоугольной области можно вычислить за один проход.

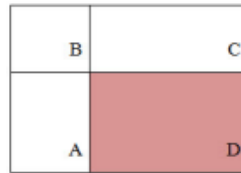


Рисунок 1.6 – Прямоугольник с областями A , B , C и D

В прямоугольнике $ABCD$ на рисунке 1.6 есть интересующий нас объект D . Сумму пикселей в прямоугольнике D можно выразить посредством суммы и разницы соседних прямоугольников по следующей формуле:

$$S(D) = I(B) + I(ABCD) - I(BA) - I(BC). \quad (1.6)$$

С помощью интегрального представления изображения можно рассчитать суммарную яркость произвольной прямоугольной области на изображении.

1.2.2 Признаки Хаара

Для обнаружения объекта в методе Виолы-Джонса используется окно определенного размера, которое перемещается по изображению, сдвигаясь каждый раз на один пиксель по горизонтали или вертикали. Признак Хаара, с помощью которого происходит поиск нужного объекта, рассчитывается в областях изображения, где проходит окно [5].

Признак – отображение $f: X \Rightarrow D_f$, где D_f – является множеством допустимых значений признака. Вектор признаков $x = (f_1(x), \dots, f_n(x))$ называется признаковым описанием объекта x , если заданы признаки f_1, \dots, f_n . На основе анализа признакового описания делается вывод о наличии или отсутствии искомого объекта в текущем окне. Наряду с этим множество $X = D_{f_1} * \dots * D_{f_n}$ называют признаковым пространством.

Признаки делятся на следующие типы в зависимости от множества D_f [12]:

- бинарный, $D_f = \{0, 1\}$;
- номинальный: D_f — конечное множество;
- порядковый: D_f — конечное упорядоченное множество;
- количественный: D_f — множество действительных чисел.

Признак Хаара вычисляется по смежным прямоугольным областям. В методе Виолы-Джонса применяются прямоугольные примитивы, изображенные на рис. 1.7.

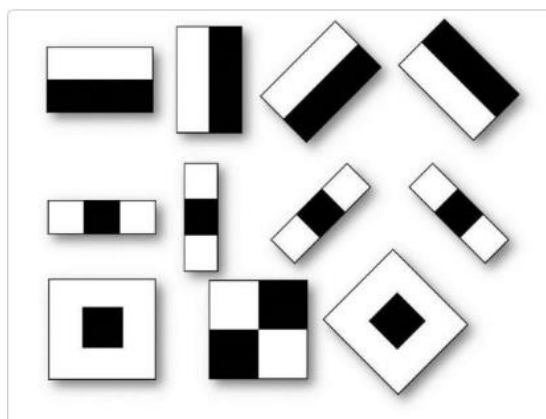


Рисунок 1.7 – Примитивы Хаара

Значение признака Хаара F вычислется по следующей формуле:

$$F = X - Y, \quad (1.7)$$

где X – суммарное значение яркостей точек, которые закрываются светлой частью, Y – суммарное значение яркостей точек, которые закрываются темной частью примитива Хаара [13]. Вычисления проводятся с помощью интегрального изображения. Рассмотренное выше, примитивы Хаара могут подсчитываться очень быстро, за постоянное время.

Признаки Хаара являются не очень хорошо подходящими для обучения и классификации, для описания объекта с достаточной корректностью требуется достаточно большее число признаков. В связи с этим признаки Хаара поступают в каскадный классификатор, служащий для быстрого отбрасывания окон, где не найден требуемый объект, и выдачи результата «истина» или «ложь» относительно нахождения объекта.

1.2.3 Классификатор на основе алгоритма адаптивного бустинга

Классификатор в задачах классификации является аппроксимирующей функцией, выносящей решение, к какой именно категории относится данный объект. Классификатор строится на основе алгоритма бустинга (от англ. boost – усиление) для того, чтобы выбрать в наибольшей степени подходящих признаков для отыскиваемого объекта на определенной области изображения.

Бустинг – это совокупность методов, содействующих увеличению точности квалификации. «Сильной» называется модель, у которых ошибки классификации допускаются в меньшем количестве. «Слабой» – модель, у которой ошибки классификации допускаются в большом количестве. Слабые модели не имеют возможности безопасно поделить классы или выдавать точные предположения. Бустинг обозначает «усиление» «слабых» моделей. Такой метод является процедурой последовательного построения композиции алгоритмов машинного обучения. В бустинге каждый следующий алгоритм устремляется компенсировать слабость композиции всех предыдущих алгоритмов. На выходе алгоритма бустинга на каждой итерации формируется обычный классификатор вида:

$$h_{j(z)} = \begin{cases} 1, & \text{если } p_i f_j(z) < p_i \theta_j \\ 0, & \text{иначе} \end{cases}, \quad (1.8)$$

где p_i – направление знака неравенства, θ_j — значение порога, $f_j(z)$ – подсчитанное значение признака, z – участок изображения [13].

Выходной классификатор имеет минимальную погрешность по отношению к текущим значениям весов, которые задействованы в процессе обучения для нахождения ошибок.

Для поиска объекта на цифровом изображении используется обученный классификатор, представленный в формате xml. Классификатор формируется на примитивах Хаара [15]. На рисунке 1.8 приведен пример структуры классификатора.

```

<maxWeakCount>6</maxWeakCount>
<stageThreshold>-1.3110191822052002e+000</stageThreshold>
<weakClassifiers>
  <_>
    <internalNodes>
      0 -1 193 1.0079263709485531e-002</internalNodes>
    <leafValues>
      -8.1339186429977417e-001 5.0277775526046753e-001</leafValues></_>
  <_>
    <internalNodes>
      0 -1 94 -2.2060684859752655e-002</internalNodes>
    <leafValues>
      7.9418992996215820e-001 -5.0896102190017700e-001</leafValues></_>

```

Рисунок

1.8 – Структура классификатора

где `<maxWeakCount>` – общее количество слабых классификаторов; `<stageThreshold>` – значение максимального порога яркости; `<weakClassifiers>` – подбор слабых классификаторов, которые выносят решение о том, что присутствует ли объект на изображении или нет; `<internalNodes>` и `<leafValues>` – характеристики определенного слабого классификатора.

На основе этого базиса строится каскад классификаторов, принимающих решение о том, распознан объект на изображении или нет. Присутствие или отсутствие объекта в окне определяется разницей между значением признака и порогом, полученным в результате обучения [15].

1.2.4 Схема обучения

Для алгоритма необходимо заранее подготовить тестовую выборку, содержащую l примеров, содержащих искомый объект и m – не содержащих. Тогда количество всех тестовых изображений будет [16]:

$$n = l + m, \quad (1.9)$$

$$X = \{ x_1, x_2, \dots, x_n \}, \quad (1.10)$$

где X – множество всех тестовых изображений, где для каждого заранее известно присутствует ли искомый объект или нет и отражено во множестве Y .

$$Y = \{ y_1, y_2, \dots, y_n \}, \quad (1.11)$$

где

$$y_1 = \begin{cases} 1, \text{ объект присутствует на изображении } x_1 \\ 0, \text{ иначе} \end{cases}. \quad (1.12)$$

Под признаком j будем понимать структуру вида

$$j = \{ \text{маска, положение, размер} \}. \quad (1.13)$$

Тогда откликом признака будет $f_j(x)$, который вычисляется как разность интенсивностей пикселей в светлой и темной областях. Слабый классификатор имеет вид:

$$h_j(x) = \begin{cases} 1, p_j f_j(x) < p_j \theta_j \\ 0, \text{ иначе} \end{cases}. \quad (1.14)$$

Задача слабого классификатора – определять присутствие объекта в больше чем 50% случаев. В методе используется процедура обучения AdaBoost, которая позволяет создавать сильный классификатор, состоящий из T слабых классификаторов и имеющий вид:

$$H(x) = \begin{cases} 1, \sum_{t=1}^T a_t h_{j(t)}(x) \geq \frac{1}{2} \sum_{t=1}^T a_t \\ 0, \text{ иначе} \end{cases}. \quad (1.15)$$

Целевая функция обучения имеет следующий вид:

$$T, h_{j(1)}, h_{j(2)}, \dots, a_1, \dots, a_T = \operatorname{argmin} \sum_{i=1}^n |H(x_i, T, h_{j(1)}, \dots, h_{j(T)}, a_1, \dots, a_T)|. \quad (1.16)$$

Перед началом обучения инициализируются веса $w_{q,i}$, где q – номер итерации, i -номер изображения.

$$w_{1,i} = \begin{cases} \frac{1}{2l}, y_i = 1 \\ \frac{1}{2m}, y_i = 0 \end{cases}. \quad (1.17)$$

После процедуры обучения получится T слабых классификаторов и T значений.

1.3 Методы сглаживания изображения

Предобработка является преобразованием исходного изображения в некоторое иное, улучшенное изображение. Безусловно, наиболее популярным способом предварительной обработки изображений является фильтрация [17], которая в большинстве приложений используется для удаления шумов. В результате фильтрации получается новое изображение того же размера, что и исходное, но содержащее значения интенсивностей пикселей, обновленных в соответствии с некоторым правилом. Существует достаточно много методов сглаживания изображений, отличающихся своими параметрами и выходными результатами. Это такие фильтры как линейные, фильтр Гаусса, нелинейные, медианные, билатеральные, ранжирующие, адаптивные, комбинированные, гибридные и др.

Цифровые изображения подвергаются воздействию различного типа шумов. Они могут возникать в процессе получения изображений, передачи информации и оцифровки данных. Процесс устранения различного типа шумов на изображении называют фильтрацией.

В процессе фильтрации значение яркости, которое признается в наименьшей степени искаженным помехой, заменяет яркостные параметры каждой точки изображения [18, 19].

Существуют частотные и пространственные методы фильтрации. Частотные методы преобразований изображений [19-24] основываются на идее Фурье преобразования. Смысл преобразования Фурье заключается в том, что исходная функция представляется в виде суммы тригонометрических функций в виде суммы разных частот, которые умножаются на заданные коэффициенты. У функции, представленной Фурье-преобразованием, после применения ряда преобразований есть возможность вернуться обратно в исходный вид. Процесс

реализации частотных методов может быть подобен пространственным методам фильтрации [18].

Пространственные методы улучшения изображений могут быть применены к растровым изображениям. В таких пространственных алгоритмах к каждой точке изображения применяются специальные операторы. Операторами являются прямоугольные матрицы или квадратные матрицы. Их называют масками, окнами или ядрами [18, 25, 26]. Во многих случаях маска представляется небольшим двумерным массивом, а методы улучшения, основывающиеся на таком аспекте, называются обработкой по маске или фильтрацией по маске.

1.3.1 Линейные фильтры

Линейные фильтры – простейшие представители пространственных методов предобработки. Линейная фильтрация сводится к пересчету значений интенсивности каждого пикселя изображения посредством вычисления взвешенной свертки интенсивностей пикселей, принадлежащих некоторой его окрестности. Размытие или сглаживание – еще один подход к предобработке. Сглаживание подобно линейной фильтрации в простейшем случае предполагает свертку с равными весовыми коэффициентами, в более сложных приложениях – свертку с дискретными значениями функции распределения Гаусса или выбор медианы среди набора интенсивностей в окрестности [27].

Отклик маски в линейной фильтрации задается суммой произведений пикселей в области накрывания фильтра. Усредняющий фильтр используется в качестве линейного сглаживающего фильтра. Выходное значение усредняющего фильтра является средним значением по окрестности маски фильтра [18]. Фильтр подобного рода используется для удаления зернистости изображения, который вызван импульсным шумом. Отклик $g(x, y)$ усредняющего фильтра, который рассчитан для фильтрации изображения f с размерами $M \cdot N$, выражается следующей формулой:

$$g(x, y) = \sum_{s=\frac{(m-1)}{2}}^{(m-1)/2} \sum_{t=\frac{(n-1)}{2}}^{(n-1)/2} w(s, t) f(x + s, y + t), \quad (1.18)$$

где $w(s, t)$ – элемент ядра свертки изображения, с размером $m \times n$; $s \in \left[-\frac{m}{2}, \frac{m}{2}\right]$ – координаты ядра по оси абсцисс, $t \in \left[-\frac{n}{2}, \frac{n}{2}\right]$ – координаты по оси ординат; $x = 0, 1, 2, \dots, M - 1$, $y = 0, 1, 2, \dots, N - 1$ – координаты изображения f .

Подобный фильтр, для удобного программного представления, выражается в следующем виде:

$$G_{i,j} = \sum_{s=\frac{(m-1)}{2}}^{(m-1)/2} \sum_{t=\frac{(n-1)}{2}}^{(n-1)/2} w_{s,t} \cdot E_{(i+s,j+t)}, \quad (1.19)$$

где $G_{i,j}$ – элемент матрицы изображения после применения фильтрации; $w_{s,t}$ – элемент массива ядра свертки изображения, с размером $m \times n$; $E_{i,j}$ – элемент матрицы изображения.

1.3.2 Фильтр Гаусса

Благодаря данному фильтру можно сгладить и уменьшить зашумленность изображения. Ядро данного фильтра можно выразить формулой:

$$F_{gauss}(i, j) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{i^2 + j^2}{2\sigma^2}\right), \quad (1.20)$$

где i, j – координаты пикселя изображения; f – сигнал, а σ – шум, находящийся на исходном изображении.

Используя фильтр Гаусса, можно удалить шум, подвергая содержательные контуры изображения размытию в малой степени. К примеру, если на исходном изображении нужно размыть мелкие детали, которые не требуют отделения от фона, а интересующие нас крупные объекты будем выделять в даль-

нейшем с помощью бинаризации.

Нелинейные фильтры

Нелинейные пространственные методы аналогичны линейным фильтрам по принципу работы. Значения элементов матрицы исходного изображения играют важную роль. Они находятся в анализируемой области, и от них зависят операции, выполняемые нелинейным фильтром. Например, работа нелинейного фильтра может заключаться в подсчёте медианы значений элементов в анализируемой области исходного изображения [18].

Медианный фильтр

При медианной фильтрации значения пикселей представляются в виде усредненных значений точек соответствующих областей. Медианный фильтр является более эффективным, чем обычный усредняющий фильтр, при устранении шумов ввиду того, что он приводит к меньшим искажениям границ объектов на изображении. Двумерная апертура с центральной симметрией используется в качестве маски. Центр двумерной апертуры помещается в фактической точке фильтрации [19]. На рисунке 1.9 показаны два примера наиболее часто применяемых вариантов окон: а) в виде креста; б) в виде квадрата. Размеры окон принадлежат к числу характеристик, которые оптимизируются в процессе анализа эффективности алгоритма. Отсчеты исходного изображения, которые оказались в пределах окна, способствуют образованию рабочей выборки текущего шага.



Рисунок 1.9 – Примеры окон при медианной фильтрации

Двумерный медианный фильтр определяется следующей формулой:

$$G_{i,j} = med[E_{(i+s,j+t)}; (s,t) \in w]; i,j \in Z^2, \quad (1.21)$$

где $G_{i,j}$ – элемент матрицы изображения после фильтрации; $w_{s,t}$ – элемент массива окан изображения, имеющий размеры $m \times n$; $E_{i,j}$ – элемент матрицы исходного изображения.

Билатеральный фильтр

Билатеральный фильтр позволяет сгладить изображение с сохранением четких границ [28].

При билатеральном фильтре интенсивность каждого пикселя вычисляется как взвешенная средняя интенсивность соседних пикселей [18]. Веса, которые связаны граничащими пикселями, изменяются. Изменение происходит в соответствии со значениями промежутка на плоскости изображения (пространственный вес) и на оси интенсивности (ранговый вес). Это приводит к сохранению четких границ объектов изображения.

В методе билатеральной решетки результаты зависят только от расстояния между пикселями. Это является основной идеей данного метода.

Изображение отображается в трехмерном пространстве. Третьим выступает координата интенсивности пикселя, которое добавляется к стандартным координатам плоскости изображения, т.е. рассчитывается однородный вектор по следующей формуле:

$$\left(\frac{I_p \cdot W_p}{W_p} \right) = \sum_{q \in S} G_{\sigma_x}(\|p - q\|) G_{\sigma_x}(|I_p - I_q|) \frac{I_q}{I}, \quad (1.22)$$

где S – множество возможных координат пикселей изображения; $p = (p_x, p_y) \in S$ – координата центрального пикселя; q – координата граничащего пикселя, $I_p, I_q \in R$ – интенсивность пикселей p и q в диапазоне $[0, 1]$; G_{σ}, G_r – Гауссовы функции с параметрами σ_s и σ_r ; W_p – суммарность весов для нормализации значений [29].

Количество граничащих пикселей задается параметром σ_s , при различности интенсивности потеря в весе соседними пикселями задается σ_r .

Произведенное пространство сжимается с установленной частотой дискретизации, при этом сохраняя информацию по каждому пикселю в соответствующей ему запоминающей ячейке. Объемную структуру, состоящую из таких ячеек, называют билатеральной решеткой. Далее выполняется обработка билатеральным фильтром, учитывая только промежуток между пикселями, в частности, свертывание с трехмерной функцией Гаусса со свойствами $\frac{\sigma_s \sigma_r}{S_s}$. В итоге, результат восстанавливается по исходным координатам пикселей: трёхлинейная интерполяция и нормализация результата. На рисунке 1.10 показан исходное изображение.



Рисунок 1.10 – Исходное изображение

На рисунке 1.11 представлены результаты сглаживания изображения с использованием однородного (а), Гауссового (б), медианного (в) и билатериального фильтра (г).

а) Однородная фильтрация



б) Гауссова фильтрация



в) Медианная фильтрация



г) Билатеральная фильтрация



Рисунок 1.11 – Результаты фильтрации изображения

По результатам можно заметить, что при применении однородного, Гауссова и медианного фильтра границы объекта существенно размываются, то есть резкость границ не сохраняется; при билатериальной фильтрации удаляются шумы и малообъемные детали, при этом резкость границ сохраняется.

1.4 Выравнивание гистограммы

Выравнивание гистограммы – это один из наиболее распространенных способов повышения контраста изображения и представляет собой метод пространственной обработки. Выходное изображение производится с равномерным

распределением интенсивности пикселя. Это означает, что гистограмма выходного изображения сглаживается и систематически расширяется [31].

Цель выравнивания состоит в том, чтобы все уровни яркости имели бы одинаковую частоту, а гистограмма соответствовала равномерному закону распределения. Допустим, что задано изображение в градациях серого, которое имеет размер $N \times M$ пикселей. Количество уровней квантования яркости пикселей (число бинов) составляет J . Тогда в среднем на каждый уровень яркости должно выпадать $n_{aver} = \frac{N \cdot M}{J}$ пикселей. Базовая математика лежит в сопоставлении двух распределений. Пусть x, y – случайные величины, описывающие изменение интенсивности пикселей на изображениях, $w_x(x)$ – плотность распределения интенсивности на исходном изображении, $w_y(y)$ – желаемая плотность распределения. Необходимо найти преобразование плотностей распределения $y = f(x)$, которое позволило бы получить желаемую плотность [31]:

$$w_y(y) = \begin{cases} \frac{1}{y_{max} - y_{min}}, & y_{min} \leq y \leq y_{max} \\ 0, & \text{в противном случае} \end{cases} \quad (1.23)$$

Обозначим через $F_x(x)$ и $F_y(y)$ интегральные законы распределения случайных величин x и y . Из условия вероятностной эквивалентности следует, что $F_x(x) = F_y(y)$. Распишем интегральный закон распределения по определению:

$$F_x(x) = F_y(y) = \int_{y_{min}}^y w_y(y) dy = \frac{y - y_{min}}{y_{max} - y_{min}}. \quad (1.24)$$

Отсюда получается, что

$$y = (y_{max} - y_{min})F_x(x) + y_{min}. \quad (1.25)$$

Осталось выяснить, как оценить интегральный закон $F_x(x)$. Для этого необходимо сначала построить гистограмму исходного изображения, затем нормализовать полученную гистограмму, разделив величину каждого бина на общее количество пикселей $N \cdot M$. Значения бинов можно

рассматривать как приближенное значение функции плотности распределения $w_x^*(x)$, $0 \leq x \leq 255$. Значение интегральной функции распределения можно представить как сумму следующего вида [18]:

$$F_x^*(x) = \sum_{j=0}^x w_x^*(j), \quad (1.26)$$

где $w_x^*(j)$ – плотности распределения.

Построенную оценку можно использовать для вычисления новых значений интенсивности. Заметим, что перечисленные преобразования гистограмм можно применять не только ко всему изображению, но и к отдельным его частям.

Алгоритм выравнивания гистограммы состоит из четырех этапов:

1. Вычисление гистограммы H исходного изображения.
2. Нормализация гистограммы. Нормализация посредством деления величины каждого бина гистограммы на общее количество пикселей.
3. Построение интегральной гистограммы.
4. Определение нового значения интенсивности пикселя

$$dst(x,y) = H'(src(x,y)), \quad (1.27)$$

где $src(x,y)$ – пиксели входного изображения; $dst(x,y)$ – пиксели выходного изображения после выравнивания.

На рисунке 1.12 представлены результаты выравнивания гистограммы после применения фильтров.

а) После однородной фильтрации

б) После Гауссовой фильтрации



в) После медианной фильтрации



г) После билатеральной фильтрации



Рисунок 1.12 – Результаты сглаживания изображения

Из результатов представленных на рисунке 1.12 можно увидеть, что выравнивание гистограммы нормализует яркость и увеличивает контрастность изображения.

1.5 Методы описания текстурных областей изображения

В цифровой обработке изображений для описания текстуры области применяются множество различных методов: статистические, структурные, фрактальные, спектральные и комбинированные (рисунок 1.13) [32].

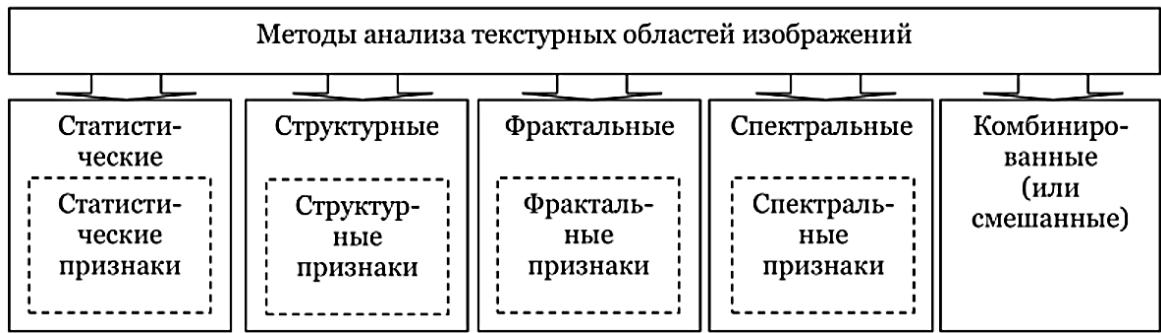


Рисунок 1.13 – Методы анализа текстурных областей изображений

Статистические методы основаны на вычислении статистических признаков текстурных областей изображения, и позволяют охарактеризовать рассматриваемую текстурную область как гладкую, грубую, зернистую (периодичную). Недостатками данных методов являются высокая вычислительная сложность и длительность процесса сегментации [18]. Общий алгоритм данного подхода можно описать в нескольких последовательных шагах [33].

Шаг 1. Считывание исходного изображения.

Шаг 2. Определение текстуры изображения.

Пусть z – является случайной величиной, которая соответствует яркостям элементов изображения, а $p(z)$, $i = 0, 1, 2, \dots, L-1$ – является ее гистограммой, где L обозначает количество разных уровней яркости. Центральный момент порядка n случайной величины z равняется [34]:

$$\mu_n(z) = \sum_{i=0}^{L-1} (z_i - m)^n p(z_i), \quad (1.28)$$

где m – среднее значение z (средняя яркость изображения) равен:

$$m = \sum_{i=0}^{L-1} z_i p(z_i). \quad (1.29)$$

Из формулы (1.29) видно, что $\mu_0 = 1$ и $\mu_1 = 0$.

Для описания текстуры изображения особенно важным является второй момент, т.е. дисперсия: $\sigma^2(z) = \mu_2(z)$. Она представляет собой меру яр-

костного контраста, которую можно применять для построения дескрипторов относительной гладкости. В формуле (24) представлена величина R , у которой для областей неизменной яркости значение равно 0 (области нулевой дисперсии) и для больших значений дисперсии $\sigma^2(z)$ она приближается к 1:

$$R = 1 - \frac{1}{1 + \sigma^2(z)}. \quad (1.30)$$

Значения дисперсии оказываются большими для полутоновых изображений со значениями элементов от 0 до 255, поэтому для использования в уравнении (1.30), разумно нормировать дисперсию до интервала $[0, 1]$. Для этого необходимо поделить значение $\sigma^2(z)$ на $(L - 1)^2$ [18].

Значение стандартного отклонения:

$$s = \sigma(z). \quad (1.31)$$

также часто используется в качестве характеристики текстуры.

Третий момент является характеристикой симметрии гистограммы:

$$\mu_3(z) = \sum_{i=0}^{L-1} (z_i - m)^3 p(z_i). \quad (1.32)$$

Для оценки текстурных особенностей используется функция энтропии, определяющая разброс в значениях яркости соседних пикселей:

$$e = - \sum_{i=0}^{L-1} p(z) \log_2 p(z_i), \quad (1.33)$$

где $p(z_i)$ вероятность текущей яркости в окрестности точки; L – количество уровней яркости; e – значение энтропии в текущей точке.

Для описания текстуры также используется мера однородности, оценивающая равномерность гистограммы:

$$U = \sum_{i=0}^{L-1} p^2(z). \quad (1.34)$$

Заметим, что текстура определяется внутри окна, скользящего по исходному изображению. При этом каждый пиксель нового изображения рассчитывается как значение текстуры, по одной из выше приведенных формул, в окрестности соответствующего пикселя исходного изображения. Т.е. гистограмма, среднее значение, дисперсия и другие величины рассчитываются заново в окрестности каждого пикселя [33].

1.6. Вывод по Главе 1

В аналитическом обзоре были рассмотрены существующие методы детектирования объектов на изображениях, такие как сравнение с шаблоном; морфологические операторы; преобразование Хафа; нахождение границ изображения с помощью алгоритмов, Робинсона, Собеля, Канни и др; метод контурного анализа; метод Виолы-Джонса и методы основанные на анализе гистограммы изображения. В результате анализа в качестве методов детектирования номерных пластин были выбраны метод контурного анализа и метод Виолы – Джонса. Эти два метода детектирования реализованы в ходе работы на магистерской диссертацией и апробированы для решения задачи детектирования автомобильных номерных знаков.

2 АЛГОРИТМ И ПРОГРАММНЫЕ СРЕДСТВА

2.1. Разрабатываемый алгоритм детектирования номерных знаков

На первом этапе работы были реализованы и сопоставлены 2 метода детектирования автомобильных номеров: метод Виолы – Джонса и метод контурного анализа. По результатам, в методе контурного анализа успешно детектировано 95 номерных знаков из 120 изображений или 79%, в методе Виолы-Джонса успешно детектировано 107 номерных знаков из 120 изображений или 89%. На основании полученных результатов для дальнейшей работы был выбран метод Виолы – Джонса. Недостатком данного метода является большое количество ложных и множественных срабатываний, в связи с этим было решено улучшать метод Виолы – Джонса. На рисунках 2.1, 2.2, соответственно, показаны блок схемы метода контурного анализа и метода Виолы – Джонса.

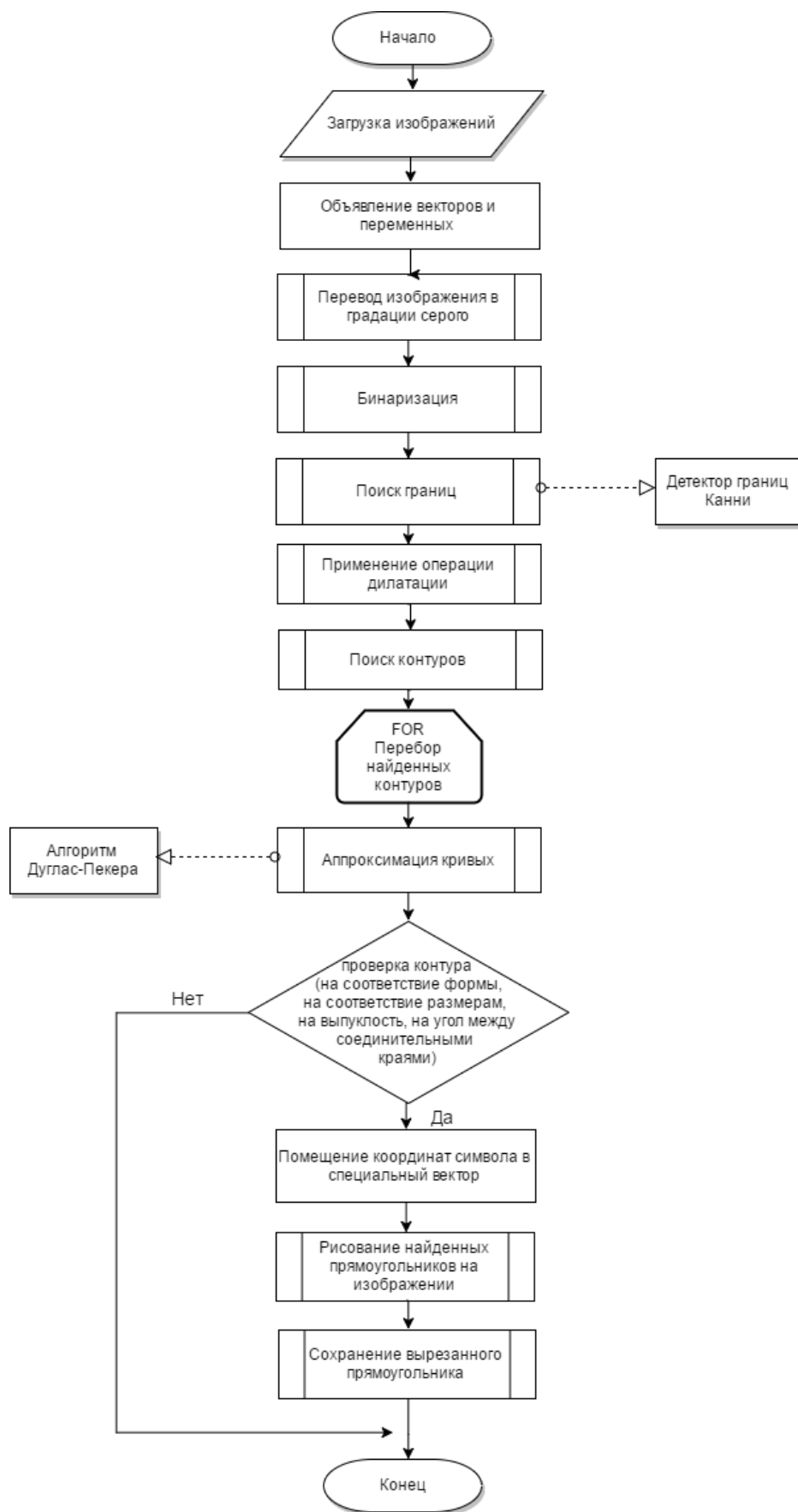


Рисунок 2.1 – Блок схема метода контурного анализа

Чтобы изображение было более приемлемо для детектирования, осуществляется его перевод в градации серого.

Так как стандартный номер пластины имеет вид прямоугольника, необходимо выделить границы существующих объектов на изображении. Для поиска краевых пикселей применяется фильтр Канни.

После поиска границ ведется бинаризация по порогу.

Далее производится дилатация. При дилатации контуры изображения утолщаются, вследствие чего удаляются возможные отверстия между краевыми сегментами.

Предлагаемый алгоритм основан на последовательном применении детектора Канни. Затем, на преобразованном изображении определяются контуры. Имеется три общих подхода к представлению контура объекта: аппроксимация кривых, прослеживание контуров и связывание точек перепадов. В данном алгоритме используется аппроксимация кривых.

Конечные точки контуров хранятся в виде горизонтальных, вертикальных и диагональных сегментов. Далее проводится проверка каждого контура на принадлежность к прямоугольнику. За основу алгоритма аппроксимации взят алгоритм Дугласа – Пекера [10]. Он приближает кривую или многоугольник с другой кривой или многоугольником с меньшим количеством вершин так, чтобы расстояние между ними было меньше или равно указанной точности.

Осуществляется фильтрация всех приближенных контуров по критериям, отражающим характеристики автомобильного номера. Площадь контура должна иметь: 4 вершины после аппроксимации и минимальную площадь больше 1000 пикселей и максимальную меньше 15 000. И должен быть выпуклым. Абсолютное значение площади используется потому, что площадь может быть положительной или отрицательной в зависимости от ориентации контура. Затем вычисляем максимальный косинус угла между соединительными краями.

В результате мы находим прямоугольную область и можем обнаружить номерную пластину с некоторой погрешностью. Рисуем прямоугольную область. Показываем вырезанный прямоугольник и сохраняем.

С помощью увеличения и уменьшения порогов можно находить, как очень маленькие номера, которые занимают 15% изображения, так и большие номера. В данном случае выставлен средний порог то есть номер занимает более 20% но не превышает 35% изображения примерно.

Общая последовательность шагов алгоритма представляется следующим образом:

1. На вход алгоритма загружается оригинальное изображение.
2. Перевод изображения в градации серого.
3. Бинаризация изображения.
4. Нахождение границ объектов.
5. Применение операции дилатации.
6. Поиск контуров на изображении.
7. Аппроксимирование кривых.
8. Далее производится проверка каждого найденного контура:
 - на соответствие формы,
 - на соответствие размерам (площади),
 - на выпуклость;
 - на угол между соединительными краями.
9. Если все условия удовлетворительны, то контур добавляется в список найденных прямоугольников.
10. Отображение найденных прямоугольных областей на изображении.
11. Сохранение номерных пластин.

Альтернативой методу контурного анализу является метод Виолы-Джонса, его блок-схема представлена на рисунке 2.2.



Рисунок 2.2 – Блок схема метода Виолы – Джонса

Алгоритм тестирования:

1. Загрузка изображения;
2. Преобразование в градации серого;
3. Загрузка каскада;
4. Поиск прямоугольников на основе каскада;
5. Рисование найденного прямоугольника.

Алгоритм детектирования ищет объекты на изображении в различных масштабах с помощью созданного каскада. Обнаруженные объекты возвращаются в виде списка прямоугольников.

К основным недостаткам метода Виолы-Джонса относится множественное и ложное детектирование, возникающее в результате резких перепадов яркости вблизи номерной пластины и за ее пределами.

Для устранения этого недостатка было принято решение модифицировать данный метод путем добавления:

- 1) предварительной обработки изображений, включающую сглаживающих фильтров и последующей эквализации гистограммы

- 2) на основе показателей текстуры сделать анализ, для того чтобы определить, какой из найденных прямоугольников среди множественно детектированных, на самом деле, является номерным знаком. На рисунке 2.3 показано блок схема модифицированного метода Виолы – Джонса.

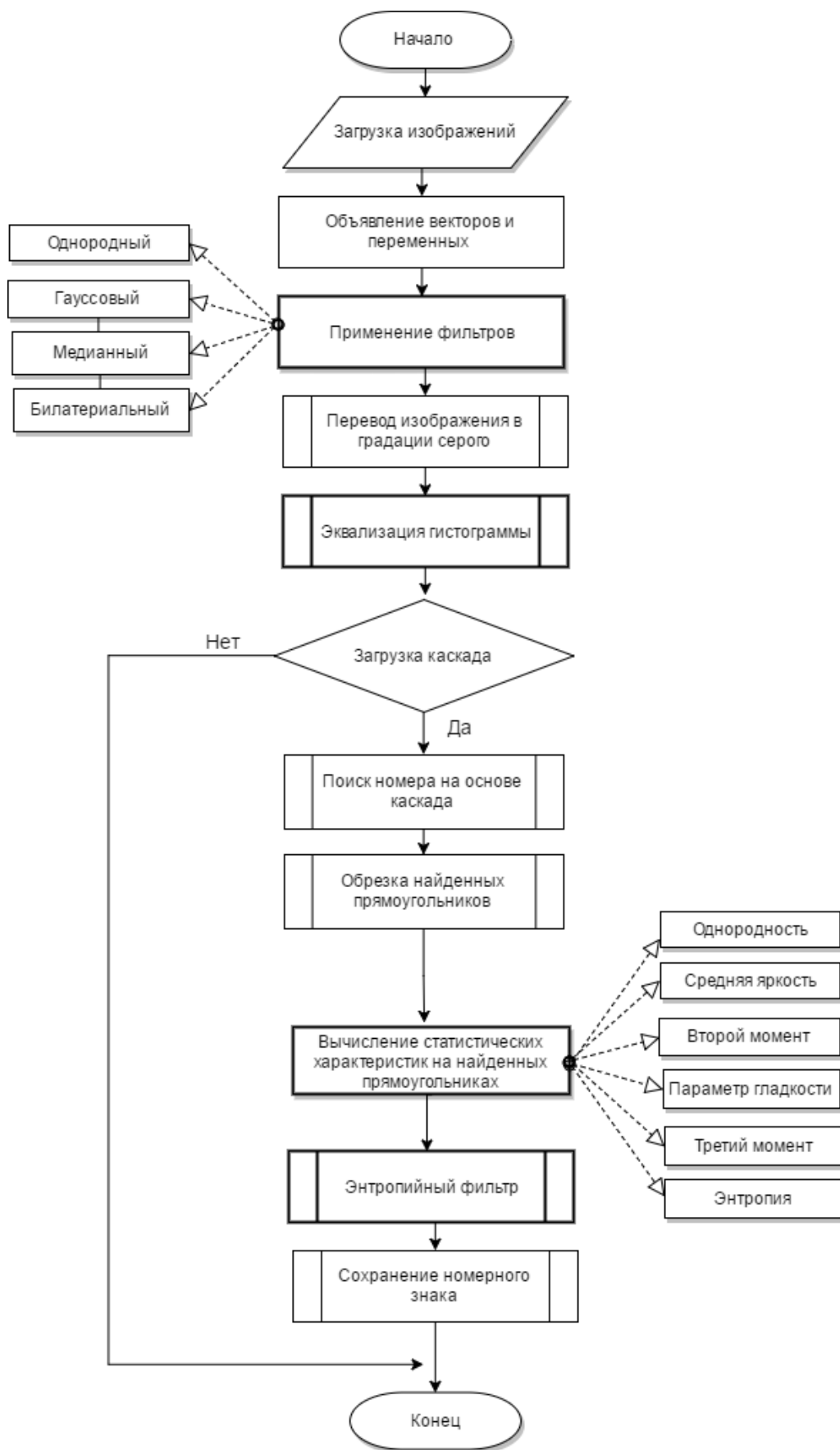


Рисунок 2.3 – Блок-схема модифицированного метода Виолы - Джонса для детектирования номерного знака

Общий алгоритм данного метода можно описать в нескольких последовательных шагах.

Шаг 1. Загрузка исходного изображения.

Шаг 2. Объявление векторов и переменных для сохранения промежуточных преобразований.

Шаг 3. Производится применение фильтра для снижения уровня шумов. В результате проведенных экспериментов наилучшие результаты детектирования получились при применении медианного фильтра.

Шаг 4. Перевод изображения в градации серого по формуле:

$$Y' = 0.2126R + 0.7152G + 0.0722B. \quad (2.1)$$

Шаг 5. Производится выравнивание гистограммы, чтобы все уровни яркости имели одинаковую частоту.

Шаг 6. Загружается каскад.

Шаг 7. Далее на основе каскада производится поиск областей, предположительно содержащих номерную пластину.

Шаг 8. Производится обрезка найденного прямоугольника.

Шаг 9. Вычисляются статистические характеристики на найденных прямоугольниках.

Шаг 10. Определяется значение энтропии и выбирается прямоугольник с наибольшим значением энтропии.

Шаг 11. Сохранение найденного номерного знака.

Шаг 11. Конец. Возвращение детектированного номерного знака.

Применение предварительной фильтрации позволит улучшить количество правильно детектированных номерных знаков. В ходе работы были сравнены следующие виды фильтрации: однородная, Гауссова, медианная и билатериальная.

Были реализованы следующие статистические характеристики текстурных особенностей: средняя яркость, дисперсия, параметр относительной гладкости R , третий момент, однородность и энтропия.

Для определения номерного знака на основе анализа выбрали энтропию. В результате проведенных экспериментов выяснилось, что остальные характеристики текстуры хуже разделяют классы «номерной знак» и «не номерной знак».

2.2 Разработка программного обеспечения

2.2.1 Язык программирования C++

Разработку программного обеспечения было решено производить на языке C++ ввиду высокой скорости работы программ, разработанных на данном языке и удобства синтаксиса.

Язык программирования предназначен для написания компьютерных программ, которые представляют собой набор правил, позволяющих компьютеру выполнить тот или иной вычислительный процесс, организовать управление различными объектами, и т. п. [35].

C++ является языком программирования общего назначения. Сочетает свойства как высокоуровневых, так и низкоуровневых языков программирования. У языка есть стандартная библиотека, которая содержит распространенные алгоритмы и контейнеры, регулярные выражения, ввод-вывод и другие возможности. Область применения – системное программирование, процедурное программирование, объектно-ориентированное программирование и обобщенное программирование.

Язык программирования C++ широко используется в разработке программных обеспечений. А именно, в создании разнообразных прикладных программ, в разработке операционных систем, драйверов устройств, а также видео игр и многое другое. Язык облегчает процесс программирования и дает возможность писать сложные и полезные программы.

Таким образом можно сказать, что язык программирования C++ является подходящим языком программирования для реализации алгоритма детектирования номерных пластин.

2.2.2 Библиотека OpenCV

OpenCV — библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом [36]. Библиотека была написана на языках C и C++ и работает на компьютерах под управлением операционных систем Linux, Windows, Mac OS X. Так же активно развиваются интерфейсы библиотеки для Python, Matlab, Ruby и других языков программирования.

Библиотека OpenCV содержит более 500 функций. Эти функции охватывают многие области компьютерного зрения, такие как: безопасность, инспекция фабричной продукции, медицина, калибровка камеры, пользовательский интерфейс, стереозрение и робототехника.

Основные модули библиотеки можно отнести к 4 группам (разделам):

- Модули `core` и `highgui` предназначены для реализации базовой функциональности (базовые структуры, ввод/вывод изображений и видео, линейная алгебра, математические функции, генераторы случайных чисел, ввод/вывод в форматах XML, YAML, быстрое преобразование Фурье и др.).
- Модули `imgproc` и `features2d` предназначены для обработки изображений (геометрические преобразования, фильтрация, сегментация, обнаружение особых точек и ребер объектов, преобразование цветовых пространств, контурный анализ и др.).
- Модули `video`, `calib3d` и `objdetect` (анализ движения и отслеживание объектов, калибровка камеры, оптический поток, вычисление положения в пространстве, детектирование объектов, построение карты глубины).
- Модуль `ml` реализует алгоритмы машинного обучения (наивный байесовский классификатор, метод ближайших соседей, бустинг, случайный лес, деревья решений, машина опорных векторов, нейронные сети и др.). На рисунке 2.4 можно увидеть основные модули библиотеки OpenCV.

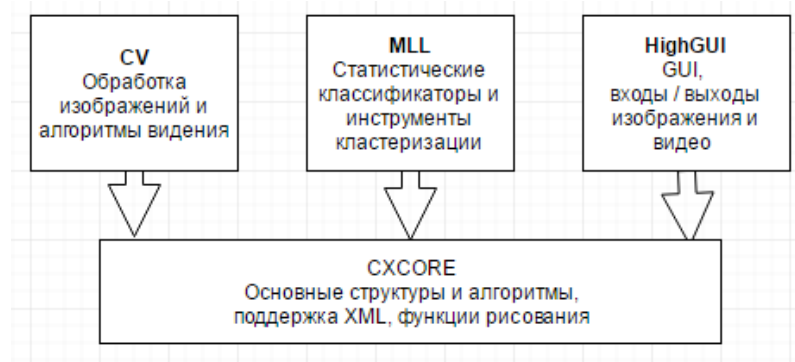


Рисунок 2.4 – Основные модули библиотеки OpenCV

Библиотека в версии 2.2 была реорганизована. Вместо универсальных модулей `sxcore`, `highGUI`, `svaux` и других было создано несколько компактных модулей с более узкой специализацией:

- `opencv_core` — модуль базовой функциональности. Содержит в себе базовые структуры, вычисления (генераторы случайных чисел, математические функции) и линейную алгебру, DCT, DFT, ввод/вывод для XML и YAML и т. д.
- `opencv_imgproc` — модуль обработки изображений (преобразование цветовых пространств, фильтрация, геометрические преобразования и т. д.).
- `opencv_highgui` — модуль простого пользовательского интерфейса, ввод/вывод изображений и видео.
- `opencv_ml` — модуль моделей машинного обучения (деревья решений, SVM, обучение со стимулированием и т. д.).
- `opencv_features2d` — модуль распознавания и описания плоских примитивов (SURF, SIFT и другие, включают специализированный фреймворк).
- `opencv_video` — модуль анализа движения и отслеживания объектов (шаблоны движения, оптический поток).
- `opencv_objdetect` — модуль обнаружения объектов на изображении (распознавание людей HOG, нахождение лиц или геометрических объектов с помощью алгоритма Виолы-Джонса и т. д.).

- `opencv_calib3d` — модуль калибровки камер, элементы обработки трёхмерных данных и поиска стерео-соответствия.
- `opencv_flann`, `opencv_contrib`, `opencv_legacy`, `opencv_gpu` [36].

OpenCV с момента своего первого выпуска в январе 1999 года, была использована во многих научно-исследовательских работах и приложениях. В том числе и в областях: наложении обычных карт и фотографий со спутника, выравнивании документов при процессе сканирования, устранении шума из медицинских изображений, анализе различных объектов, в системах безопасности, автоматическое наблюдение, в системах контроля качества на производстве, калибровки видеокамер, а также в беспилотных летательных, наземных и подводных аппаратах.

Как следует из анализа, библиотека OpenCV является достаточно универсальным инструментом для работы с изображениями и видео, поэтому было принято решение использовать именно ее для реализации работы.

2.3 Описание разработанного ПО

В таблице 2.1 представлены функции разработанного программного обеспечения.

Таблица 2.1 – Описание функций разрабатываемого ПО

Название функции	Описание функции
<pre>static void findSquares(const Mat& image, vector<vector<Point>> &squares)</pre>	<p>Реализация поиска номерной пластины</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>const Mat& image</i> — исходное изображение, представленное в виде матрицы со значениями яркости пикселей; 2) <i>vector<vector<Point>> &squares</i> — вектор содержащий координаты прямоугольников. <p>Функция возвращает последовательность прямоугольников, обнаруженных на изображении. По-</p>

	<p>следовательность хранится в виде вектора.</p>
<p>static double angle(Point pt1, Point pt2, Point pt0)</p>	<p>Функция находит косинус угла между векторами.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>Point pt1</i> – точка (структура из двух переменных (x,y)); 2) <i>Point pt2</i> – точка (структура из двух переменных (x,y)); 3) <i>Point pt0</i> – точка (структура из двух переменных (x,y)). <p>Функция возвращает значение угла.</p>
<p>void Canny(InputArray image, OutputArray image, double threshold1, double threshold2, int apertureSize)</p>	<p>Функция производит поиск границ используя алгоритм Канни.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>InputArray highestEntropySegment</i> – изображение в оттенках серого; 2) <i>OutputArray highestEntropySegment</i> – изображение для хранения границ, найденных функцией. Он имеет тот же размер и тип, что и входное изображение; 3) <i>double 50</i> – порог минимума; 4) <i>double 200</i> – порог максимума; 5) <i>int 3</i> – размер ядра для оператора Собеля. <p>Функция возвращает найденные ребра в изображении.</p>
<p>dilate(InputArray image, OutputArray image, InputArray kernel, Point anchor=Point (-1,-1))</p>	<p>Операция дилатации. Расширяет изображение с помощью определенного структурирующего элемента.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>InputArray gray</i> – входное изображение; 2) <i>OutputArray gray</i> – выходное изображение того же размера и типа, что и входное; 3) <i>InputArray Mat()</i>– структурирующий элемент, используемый для расширения; 4) <i>Point Point(-1, -1)</i> – положение якоря внутри элемента; Значение по умолчанию (-1, -1) озна-

		<p>чает, что привязка находится в центре элемента.</p> <p>Функция возвращает обработанное изображение.</p>
<p>void findContours (InputOutputArray image, OutputArrayOfArrays contours, int mode, int method)</p>		<p>Находит контуры в бинарном изображении.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>InputOutputArray highestEntropySegment</i> – исходное изображение; 2) <i>OutputArrayOfArrays contours</i> – обнаруженные контуры. Каждый контур сохраняется как вектор точек; 3) <i>int RETR_LIST</i> – режим поиска контуров. Извлекает все контуры без установления каких-либо иерархических связей; 4) <i>int CHAIN_APPROX_SIMPLE</i> – метод контурной аппроксимации. Сжимает горизонтальные, вертикальные и диагональные сегменты и оставляет только их конечные точки. <p>Функция возвращает извлеченные контуры.</p>
<p>void approxPolyDP(InputArray curve, OutputArray approxCurve, double epsilon, bool closed)</p>	<p>ар- dou- ble</p>	<p>Аппроксимирует полигональную кривую (ы) с указанной точностью. Он использует алгоритм Дугласа-Пьюкера.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>InputArray Mat(contours[i])</i> – входной вектор двумерной точки, сохраненный в Mat; 2) <i>OutputArray approx</i> – Результат аппроксимации; 3) <i>double arcLength(Mat(contours[i]), true)*0.02</i> – параметр, определяющий точность аппроксимации. Это максимальное расстояние между исходной кривой и ее приближением; <p><i>arcLength</i> – вычисляет периметр контура.</p> <ol style="list-style-type: none"> 1) <i>InputArray Mat(contours[i])</i> – входной вектор двумерных точек, сохраненный в Mat; 2) <i>bool True</i> – флаг, указывающий, закрыта ли кривая или нет.

	<p>4) <i>bool True</i> – аппроксимированная кривая закрыта (ее первая и последняя вершины связаны).</p> <p>Функция возвращает аппроксимированные кривые.</p>
<p><code>bool isContourConvex (InputArray contour)</code></p>	<p>Функция проверяет, является ли входной контур выпуклым или нет.</p> <p>Входные аргументы:</p> <p>1) <i>InputArray Mat(approx)</i> – входной вектор двумерных точек (контурные вершины), хранящиеся в <i>Mat</i>.</p> <p>Функция возвращает вектор контуров.</p>
<p><code>double contourArea(InputArray contour)</code></p>	<p>Вычисляет площадь контура по формуле Грина.</p> <p>Входные аргументы:</p> <p>1) <i>InputArray Mat(approx)</i> – входной вектор двумерных точек (контурные вершины), хранящиеся в <i>Mat</i>.</p> <p>Функция возвращает значение площади контура.</p>
<p><code>void calcHist(const Mat* images, int nimages, const int* channels, InputArray mask, OutputArray hist, int dims, const int* histSize, const float** ranges, bool uniform=true, bool accumulate=false)</code></p>	<p>Вычисляет гистограмму массива изображений.</p> <p>Входные аргументы:</p> <p>1) <i>const Mat &plateSegment</i> – исходный массив;</p> <p>2) <i>int I</i> – количество исходных изображений;</p> <p>3) <i>const int 0</i> – список каналов, используемых для вычисления гистограммы;</p> <p>4) <i>InputArray Mat()</i> – булева маска;</p> <p>5) <i>OutputArray Hist</i> – выходная гистограмма, соответствующего размера (т.е. размерности, равной числу плоскостей изображения передаваемых изображений);</p> <p>6) <i>int I</i> – размерность гистограммы;</p> <p>7) <i>const int &histSize</i> – массив размеров гистограммы в каждом измерении;</p> <p>8) <i>const float ranges</i> – массив массивов границ гистограммы в каждом измерении;</p> <p>9) <i>bool true</i> – флаг, указывающий, равномерна ли</p>

	<p>гистограмма или нет;</p> <p>10) <i>bool false</i> – флаг накопления.</p> <p>Функции возвращает гистограмму изображения.</p>
<p>void cvSetImageROI(IplImage* image, CvRect rect)</p>	<p>Устанавливает область интереса (ROI) изображения для заданного прямоугольника.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>IplImage* segment</i> – указатель на участок изображения; 2) <i>CvRect boundingRect(squares[s])</i> – прямоугольник ROI. <p>Функции возвращает заданную область изображения</p>
<p>static void drawSquares (Mat& highestEntropySegment, const vector<vector<Point>>&squares)</p>	<p>Функция рисует все квадраты на изображении.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>Mat& highestEntropySegment</i> – исходное изображение; 2) <i>const vector<vector<Point>>&squares</i> – вектор хранящий координаты прямоугольников. <p>Функция возвращает обработанное изображение с нарисованными прямоугольниками.</p>
<p>void polylines(highestEntropySegment, &p, &n, 1, true, Scalar(0, 255, 0), 3, CV_AA)</p>	<p>Рисует несколько полигональных кривых.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>Mat&highestEntropySegment</i> – входное изображение; 2) <i>const Point** &p</i> – массив полигональных кривых; 3) <i>const int* &n</i> – массив многоугольных счетчиков вершин; 4) <i>int 1</i> – количество кривых; 5) <i>bool true</i> – флаг, указывающий, закрыты ли рисованные полилинии или нет. Если они закрыты, функция рисует линию от последней вершины каждой кривой до ее первой вершины; 6) <i>const Scalar(0, 255, 0)</i> – цвет полилинии;

	<p>7) <i>int</i> 3 –толщина кромки полилинии;</p> <p>8) <i>int</i> CV_AA – тип сегментов линии. Сглаженная линия.</p> <p>Функция возвращает нарисованных кривых на изображении.</p>
cvtColor(InputArray dst , OutputArray dst , int code)	<p>Преобразует цветное изображение в изображение в оттенках серого.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>InputArray dst</i> – входное изображение; 2) <i>OutputArray dst</i> – выходное изображение; 3) <i>COLOR_BGR2GRAY</i> – код преобразования цветового пространства. <p>Функция возвращает обработанное изображение.</p>
equalizeHist(InputArray dst , OutputArray dst)	<p>Функция выравнивает гистограмму изображения в градации серого.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>dst</i> – входное изображение; 2) <i>dst</i> – выходное изображение. <p>Функция возвращает обработанное изображение.</p>
bool CascadeClassifier.load (const string& filename)	<p>Загружает классификатор из файла</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>"haarcascade_russian_plate_number.xml"</i> - Имя файла, из которого загружается классификатор.
cascadeSym- bol.detectMultiScale(const Mat& image , vector<Rect>& objects)	<p>Обнаруживает объекты разных размеров во входном изображении. Обнаруженные объекты возвращаются в виде списка прямоугольников.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>const Mat& dst</i> – Матрица типа CV_8U, содержащая изображение, в котором обнаружены объекты; 2) <i>vector<Rect>& symbols</i> – Вектор прямоугольников, где каждый прямоугольник содержит обнаруженный объект.

<p>Mat imread(files.at(curImg), 1)</p>	<p>Функция загружает изображение из указанного файла и возвращает его.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>const string& files.at(curImg)</i> – имя загружаемого файла; 2) <i>int I</i> – указывает на тип цвета загруженного изображения. <p>Функция возвращает исходное изображение.</p>
<p>void imshow(const string* name, InputArray img)</p>	<p>Отображает изображение в указанном окне.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>const string& "Original"</i> – название окна; 2) <i>InputArray src</i> – изображение для показа. <p>Функция возвращает исходное изображение.</p>
<p>Mat imwrite(const string* filename, InputArray img)</p>	<p>Сохраняет изображение в указанный файл.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>const char& "test/"</i> – имя файла, где будет храниться выходное изображение; 2) <i>InputArray dst</i> – изображение для сохранения.
<p>Функции для расчета статистических характеристик</p>	
<p>float entropy(Mat seq, Size size, int index)</p>	<p>Функция рассчитывающая энтропию изображения.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>Mat seq</i> – вычисление гистограммы; 2) <i>Size size</i> – размер(ширина, высота) изображения или прямоугольника; 3) <i>int index</i> – количество различных уровней яркости. <p>Функция возвращает значение энтропии изображения.</p>
<p>float homogenous(Mat seq, Size size, int index)</p>	<p>Функция рассчитывающая однородность изображения.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>Mat seq</i> – вычисление гистограммы; 2) <i>Size size</i> – размер(ширина, высота) изображе-

	<p>ния или прямоугольника;</p> <p>3) <i>int index</i> – количество различных уровней яркости.</p> <p>Функция возвращает значение однородности изображения.</p>
float avbrightness(Mat seq, Size size, int index)	<p>Функция рассчитывающая среднюю яркость изображения.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>Mat seq</i> – вычисление гистограммы; 2) <i>Size size</i> – размер(ширина, высота) изображения или прямоугольника; 3) <i>int index</i> – количество различных уровней яркости. <p>Функция возвращает значение средней яркости изображения.</p>
float secondMoment(Mat seq, Size size, int index)	<p>Функция рассчитывающая второй момент изображения.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>Mat seq</i> – вычисление гистограммы; 2) <i>Size size</i> – размер(ширина, высота) изображения или прямоугольника; 3) <i>int index</i> – количество различных уровней яркости. <p>Функция возвращает значение второго момента изображения.</p>
float Normir(Mat seq, Size size, int index)	<p>Функция рассчитывающая параметр относительной гладкости изображения.</p> <p>Входные аргументы:</p> <ol style="list-style-type: none"> 1) <i>Mat seq</i> – вычисление гистограммы; 2) <i>Size size</i> – размер(ширина, высота) изображения или прямоугольника; 3) <i>int index</i> – количество различных уровней яркости.

	Функция возвращает значение параметра относительной гладкости изображения.
float thirdMoment(Mat seq, Size size, int index)	Функция рассчитывающая третий момент изображения. Входные аргументы: 1) <i>Mat seq</i> – вычисление гистограммы; 2) <i>Size size</i> – размер(ширина, высота) изображения или прямоугольника; 3) <i>int index</i> – количество различных уровней яркости. Функция возвращает значение третьего момента изображения.
Функции для фильтрации	
void blur(InputArray src, OutputArray dst, Size ksize)	Функция выполняет однородную фильтрацию. Входные аргументы: 1) <i>InputArray dst</i> – входное изображение; 2) <i>OutputArray dst</i> – выходное изображение; 3) <i>Size(3, 3)</i> – размер ядра. Функция возвращает обработанное изображение.
void GaussianBlur(InputArray src, OutputArray dst, Size ksize)	Функция выполняет Гауссовую фильтрацию. Входные аргументы: 1) <i>InputArray src</i> – входное изображение; 2) <i>OutputArray src</i> – выходное изображение; 3) <i>Size(5, 5)</i> – размер ядра. Функция возвращает обработанное изображение.
void medianBlur(InputArray src, OutputArray dst, Size ksize)	Функция выполняет медианную фильтрацию. Входные аргументы: 1) <i>InputArray src</i> – входное изображение; 2) <i>OutputArray dst</i> – выходное изображение; 3) <i>3</i> – размер ядра. Функция возвращает обработанное изображение.
void bilateralFilter(InputArray src, OutputArray dst, Size ksize, int d)	Функция выполняет билатериальную фильтрацию. Входные аргументы:

	<p>1) <i>InputArray src</i>– входное изображение;</p> <p>2) <i>OutputArray src</i> – выходное изображение;</p> <p>3) 2, 5*2, 5/2 – размер ядра.</p> <p>Функция возвращает обработанное изображение.</p>
--	---

2.4 Выводы по Главе 2

Поставленная задача была реализована в среде Microsoft Visual Studio 2013 на языке программирования C++ с использованием библиотеки компьютерного зрения с открытым исходным кодом OpenCV. Исходными данными являются изображение с автомобильными номерами.

Были реализованы методы контурного анализа и Виолы-Джонса, функции предварительной обработки изображения и статистические методы описания текстурных областей изображения.

Программное обеспечение позволяет детектировать автомобильные номерные пластины на изображениях. Выходными данными программы являются изображения детектированных номерных пластин.

3 РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ

Для тестовой выборки была подготовлена база состоящая из 120 изображений с номерными знаками. Эти изображения были сняты в различных условиях освещения и разделены на 4 группы: номерная пластина находится под углом наклона меньше 15°, номерная пластина находится под углом наклона больше 15°, номерная пластина загрязнена и снята при плохом освещении.

Для обучения классификатора Виолы – Джонса было построено обучающее множество из изображений, содержащих и не содержащих автомобильных номерных знаков.

В таблице 3.1 представлены конфигурации компьютера, на котором проводилось тестирование.

Таблица 3.1 – конфигурации компьютера

Название параметра	Характеристика
Производитель	Lenovo
Модель	B50-70
Операционная система	Microsoft Windows 10
Тип системы	64 – разрядная операционная система, процессор x64
Центральный процессор (ЦП)	Intel Core(TM) i74510U
Частота процессора	2.00 ГГц
Оперативная память	8 Гб
Видеокарта	AMD Radeon R5 230
Объем видеопамяти	512 Мб

3.1 Сравнение метода контурного анализа и метода Виолы - Джонса

Результаты сопоставления методов контурного анализа и Виолы Джонста при различных условиях съемки представлены на рисунках 3.1 – 3.4.

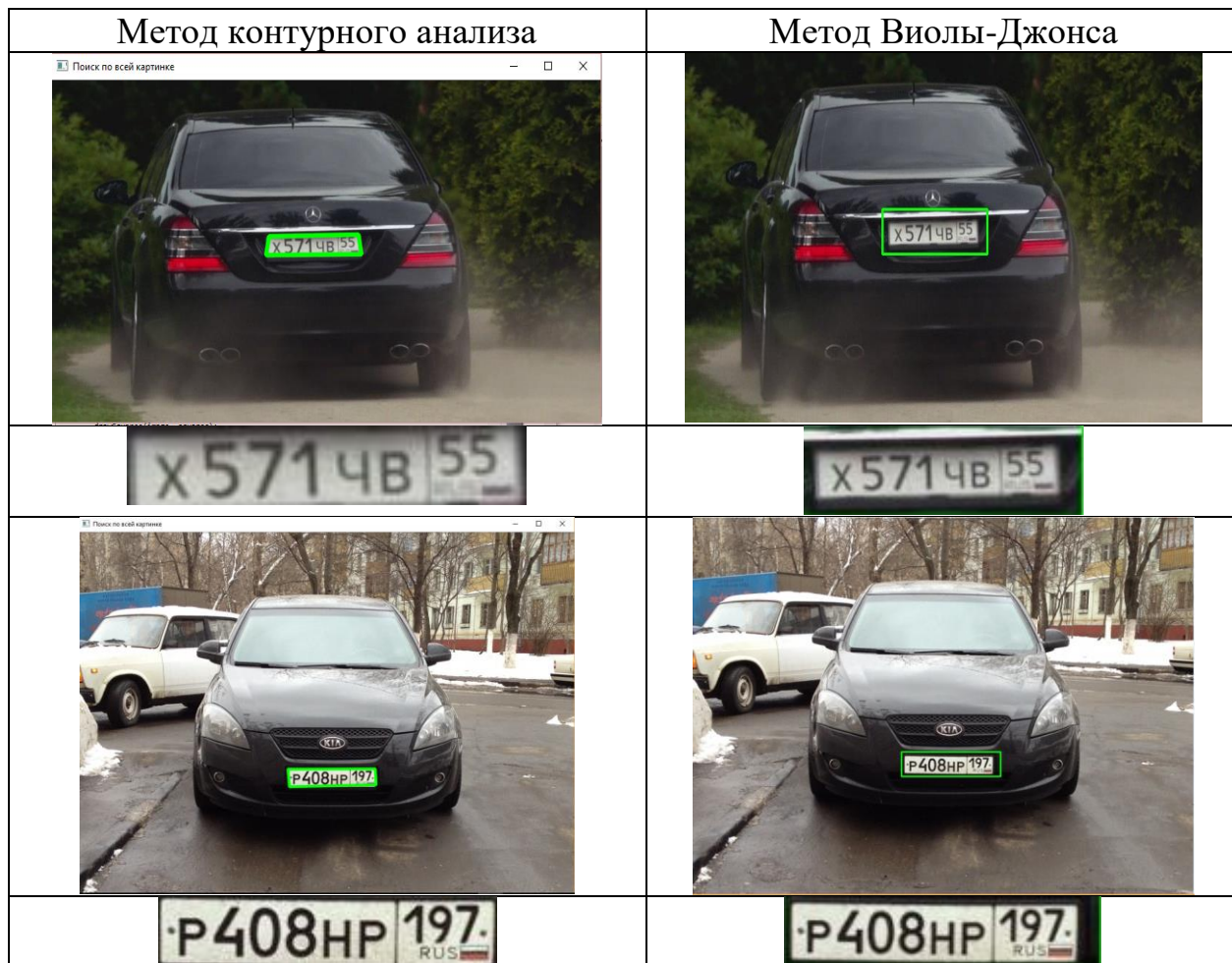


Рисунок 3.1 – Результаты детектирования номерной пластины под углом наклона меньше 15°





Рисунок 3.2 – Результаты детектирования номерной пластины под углом наклона больше 15°

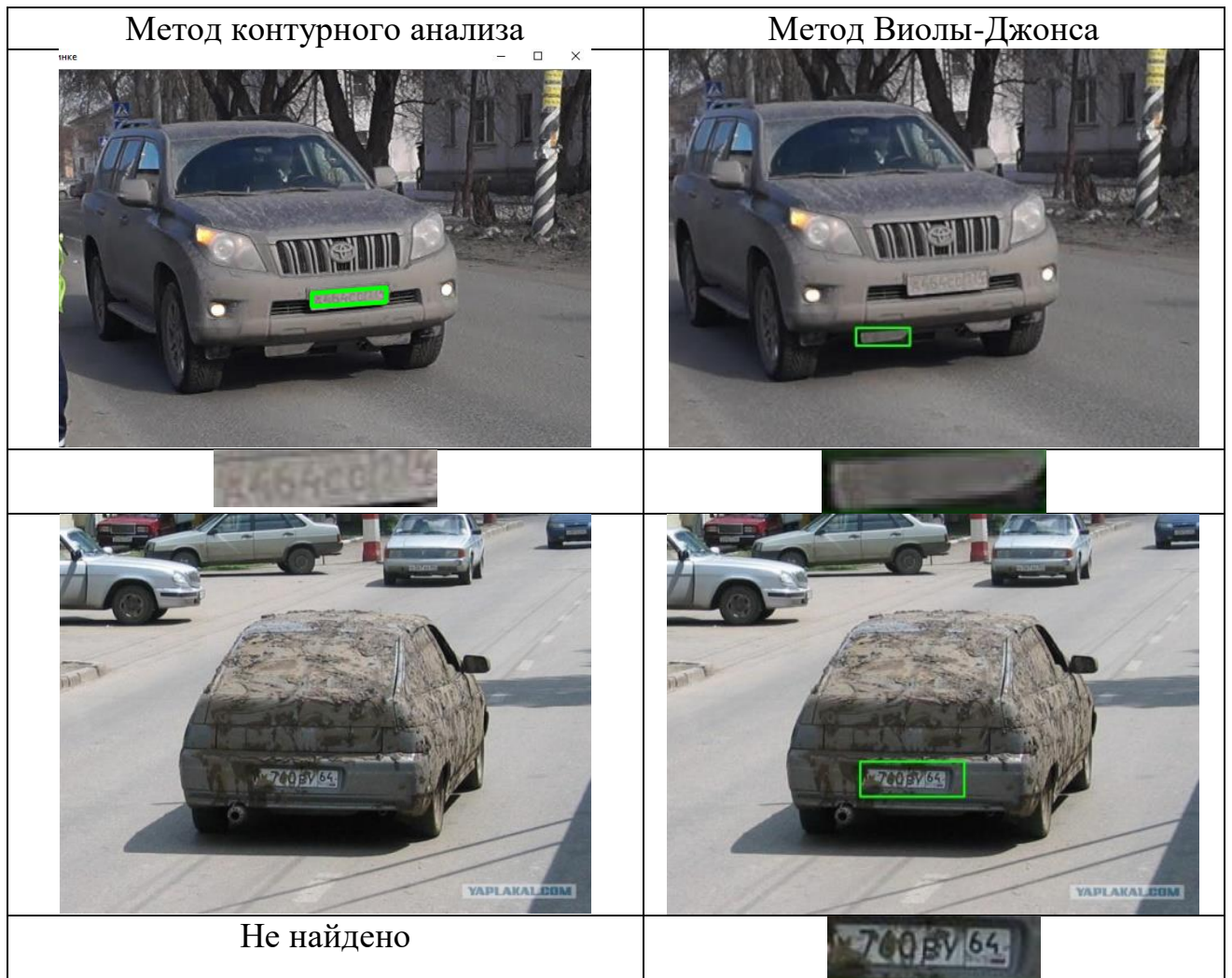


Рисунок 3.3 – Результаты детектирования загрязненной номерной пластины

Метод контурного анализа	Метод Виолы-Джонса
	
Не найдено	
	
	

Рисунок 3.4 – Результаты детектирования номерной пластины при плохом освещении

В таблице 3.1 показаны результаты детектирования номерных пластин для разных групп с применением различных фильтров. Значения столбцов:

- «+» - количество правильно детектированных номерных знаков,
- «-» - количество не детектированных номерных пластин,
- «м» - количество множественно детектированных.

Если имеет место множественное обнаружение одной номерной пластины, описываемые областями разной площади, и еще другие области близи номерной пластины, то они понимаются как «множественные». Пример множественного детектирования приведен в рисунке 3.5. Далее необходимо выяснить, какая из детектированных областей наиболее точно включает номерной знак.



Рисунок. 3. 5 – Пример множественного детектирования

Таблица 3.1 – Результаты детектирования номерных пластин методом контурного анализа и методом Виолы-Джонса при различных условиях

	Группы	Метод контурного анализа			Метод Виолы-Джонса		
		«+»	«-»	«M»	«+»	«-»	«M»
а)	номерная пластина находится под углом наклона меньше 15°	62	8	0	70	0	16
б)	номерная пластина находится под углом наклона больше 15 °	29	1	0	28	2	11
с)	номерная пластина загрязненная	1	9	0	2	8	1
д)	при плохом освещении	3	7	0	7	3	2

В таблице 3.2 показан общий результат детектирования.

Таблица 3.2. Общие результаты детектирования

Название метода	Правильно детектированные, количество	Не детектированные, количество	Множественно детектированные, количество
Метод контурного анализа	95	25	0
Метод Виолы-Джонса	107	13	30

Точность детектирования определялась по формуле:

$$\text{Точность детектирования} = \frac{y}{x} * 100\%. \quad (3.1)$$

Общее количество изображений $x = 120$

Количество правильно детектированных: $y1 = 95$

$y2 = 107$

где $y1$ – количество правильно детектированных для метода контурного анализа; $y2$ – количество правильно детектированных для метода Виолы-Джонса.

Итоговая точность верного детектирования номерного знака представлена в таблице 3.3.

Таблица 3.3. Точность детектирования методов детектирования номерных знаков

Методы	Точность детектирования
Метод контурного анализа	79 %
Метод Виолы-Джонса	89 %

При детектировании автомобильных номеров метод Виолы-Джонса показал лучшие результаты точности 89% по сравнению с методом контурного анализа – 79%.

3.2 Тестирование предложенного метода детектирования

На выборке из 120 изображений было проведено тестирование предложенного метода Виолы – Джонса с применениями различных фильтров. Результаты тестирования метода на соответствующих группах изображений (рисунки 3.6 - 3.9):



Рисунок 3.6 – Результат детектирования номерной пластины под углом наклона меньше 15°



Рисунок 3.7 – Результат детектирования номерной пластины под углом наклона больше 15°



Рисунок 3.8 – Результаты детектирования загрязненной номерной пластины



Рисунок 3.9 – Результаты детектирования номерной пластины при плохом освещении

В таблице 3.4 показаны результаты детектирования для разных групп с применением различных фильтров. Значения столбцов:

- «+» - количество правильно детектированных номерных знаков,
- «-» - количество не детектированных номерных пластин,
- «М» - количество множественно детектированных.
- а) - номерная пластина находится под углом наклона меньше 15°
- б) - номерная пластина находится под углом наклона больше 15°
- с) - номерная пластина загрязненная
- д) - при плохом освещении

Таблица 3.4 – Результаты детектирования для разных групп

Гр.	Общ. кол. изобр	До предобработки			Однородный фильтр			Фильтр Гаусса			Медианный фильтр			Билатеральный фильтр		
		«+»	«-»	«М»	«+»	«-»	«М»	«+»	«-»	«М»	«+»	«-»	«М»	«+»	«-»	«М»
а	70	70	0	16	70	0	16	70	0	18	70	0	17	70	0	17
б	30	28	2	11	27	3	11	28	2	13	28	2	8	28	2	10
с	10	2	8	1	3	7	1	4	6	2	4	6	1	5	5	1
д	10	7	3	2	9	1	0	9	1	1	8	2	0	9	1	1
О б щ	120	107	13	30	109	11	28	111	9	34	110	10	26	112	8	30

Из таблицы 3.4 можно сделать следующие выводы:

1. при расположении номерной пластины под углом наклона меньше 15° по сравнению с другими фильтрами хороший результат показало применение однородного фильтра;

2. при расположении номерной пластины под углом наклона больше 15° результаты всех фильтров, кроме однородного фильтра были равны результатам до предобработки;

3. для загрязненных номерных пластин билатеральный фильтр показал хороший результат, так как увеличилось количество правильно детектированных номерных пластин;

4. для номерных пластин при плохом освещении результаты были схожими. Множественных детектировании не стало при применении однородного фильтра .

Точность детектирования определялась по формуле:

$$\text{Точность детектирования} = \frac{y}{x} * 100\%. \quad (3.2)$$

где x – общее количество изображений; y – количество правильно, множественно и не детектированных номерных пластин для каждого фильтра.

В таблице 3.5 показаны общие результаты и точности детектировании для различных фильтров.

Таблица 3.5 –Общие результаты и точности детектировании

Метод Виолы - Джонса	Правильно детектированные		Не детектированные		Множественное детектирование	
	Кол-во	%	Кол-во	%	Кол-во	%
До предобработки	107	89,17	13	10,83	30	25
С применением однородного фильтра	109	92,5	11	9,17	28	23,3
С применением фильтра Гаусса	111	92,5	9	7,5	34	28,3
С применением медианного фильтра	110	91,67	10	8,33	26	21,6
С применением билатерального фильтра	112	93,33	8	6,67	30	25

Из результатов таблицы 2 можно увидеть, что после предобработки количество детектированных номерных пластин стало больше, чем до предобработки. По количеству правильно детектированных самый лучший



результат показал билатеральный фильтр. Все дальнейшие эксперименты проводились с применением данного фильтра.

3.2.1 Вычисление статистических характеристик текстур

После предобработки изображения на найденных прямоугольниках вычислялось следующие статистические характеристики текстур: средняя яркость, дисперсия, параметр относительной гладкости R , третий момент, однородность и энтропия. Наиболее вероятный номерной знак определялся по этим значениям.

В таблице 3.6 приведены примеры значения описанных выше характеристик для текстур двух видов: номерной знак и не номерной знак.

Таблица 3.6 – Результат расчета двух типов текстур

№	Тип текстуры	Среднее	Станд. отклонение	R (норм)	Третий момент	Однородность	Энтропия	Изображение
1	Номер	106.095	11251.9	255.977	-1.19354	0.0199	6.65551	
	Не номер	125.355	15710.2	255.984	-1.96912	0.0143	6.12166	

Несмотря на то, что среднее значение и стандартное отклонение показали наибольшую разницу при оценке номерного и неномерного знака, внутриклассовая оценка данных характеристик очень сильно меняется. Несмотря на то, что значения энтропии для неномерного и номерного знака отличаются незначительно, внутриклассовое различие у этой характеристики невелико. В реализованной программе номерной знак определяется по наибольшему значению энтропии. Выбор наибольшего значения обусловлен тем, что номерная пластина, как правило, более контрастна, чем область, не являющаяся номером, так как содержит черные символы на светлом фоне. Как видно из таблицы 3 значение энтропии у номерного знака больше, чем у не номерного знака.

В таблице 3.7 приведены результаты тестирования классификации

множественных номерных знаков на основе оценки текстурных особенностей энтропии.

Таблица 3.7 – Результаты оценки текстурных особенностей

Общее количество множественных срабатываний	100%
Число правильно определенных номерных знаков	66,67%
Число неправильно определенных номерных знаков	33,33%

Из найденных прямоугольников на изображении с помощью анализа текстурных особенностей энтропии нам удалось правильно определить 66,67% номерных знаков и снизить количество множественных срабатываний.

3.3 Результаты экспериментальной части

Итогового результата после предобработки и анализа характеристик текстуры можно увидеть в таблице 3.8.

Таблица 3.8 – Итоговые результаты

Предложенный метод	Правильно детекти- рованные		Не детектирован- ные		Множественное детектирование	
	Кол-во	%	Кол-во	%	Кол-во	%
До предобработки	107	89,17	13	10,83	30	25
С применением билатериального фильтра	112	93,33	8	6,67	30	25
После анализа текстур	112	93,33	8	6,67	10	8,33

После предобработки и анализа характеристик текстуры количество правильно детектированных номерных знаков увеличилось, а количество множественных срабатываний уменьшилось.

В общем случае, после предобработки номерные пластины которые не детектировалась ранее, были детектированы, таким образом увеличили

количество детектированных номерных пластин увеличилось. Правильно детектированные номерные знаки до предобработки составляли 89,17%, а после предобработки 93,33 %.

На основе текстурных особенностей из найденных прямоугольников на одном изображении смогли определить наиболее вероятный номерной знак и снизить появление множественных срабатываний на 66,67%.

3.4 Сравнение алгоритмов

Разработанный в рамках выпускной квалификационной работы алгоритм предназначен для детектирования автомобильных номеров на изображениях.

Функциональным аналогом разработанной системе являются такие системы, как:

1. Автоматическое распознавание номерных знаков на основе преобразовании Хафа – «ANPR», описан в статье [37].
2. Обнаружение и распознавание номерных пластин автомобиля с помощью вейвлет преобразования – «VLPR», описан в статье [38],
3. Обнаружение номерных знаков с использованием метода Виолы Джонса – «LPR», описан в статье [39].

Эти алгоритмы позволяют детектировать и распознавать автомобильные номерные знаки на изображениях. Коммерческие программы являются закрытыми, и единственной характеристикой, которую можно оценить, является итоговая точность детектирования автомобильного номера, которая

часто является преувеличенной. Точности верного детектирования предлагаемых алгоритмов приведены в таблице 3.9.

Таблица 3.9 – Точности алгоритмов детектирования автомобильных номеров

	Алгоритмы	Уровень верного детектирования
1	«ANPR»	94,11%
2	«VLPR»	88,71%
3	«LPR»	91,54%
4	Разработанный алгоритм	93,33%

База изображений для системы «ANPR» состоит из 102 изображений, которые были взяты из разных сцен при различных условиях освещения. Предлагаемый метод успешно находит 96 номерных знаков и количество не найденных номерных знаков 6. Таким образом, точность успешного детектирования составляет 94,11%.

База изображений для системы «VLPR» состоит из 257 изображений. Количество успешно найденных номерных знаков 228, а не найденных 29. Точность успешного детектирования составляет 88,71%.

Для системы «VLPR» общее количество тестовых изображений составляет 110 изображений и точность успешного детектирования составляет 91,54%.

База изображений для разработанного алгоритма составляет 120 изображений. Успешно детектированные номерные знаки 112, а не детектированные 8. Точность успешного детектирования составляет 93,33%.

Коммерческая программа «ANPR» на основе преобразований Хафа показала более высокую точность детектирования, а программы «VLPR» на основе вейвлет преобразования – более низкие результаты.

Разработанный алгоритм, в котором применены предложенные в выпускной квалификационной работе алгоритмы детектирования, показал

хорошую точность детектирования. В отличие от алгоритма «LPR», разработанный алгоритм позволяет изменять размеры изображения и не чувствителен к входному формату данных.

Таким образом, разработанный алгоритм является более гибким и охватывает большее количество областей применения при более высокой точности детектирования.

4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

Данная работа посвящена разработке и программной реализации алгоритма детектирования номерных знаков на изображениях. Как и любой проект, вне зависимости от того какой характер он несет, научный или практический, работа имеет экономическую составляющую, которую необходимо оценить с целью предоставления полной картины значимости осуществленной работы и ее вклада в заданную предметную область.

Целью данного раздела является оценка и анализ экономических аспектов данной работы. Основными критериями для оценки служат эффективность проекта, затраты на его выполнение, а также перспективы его внедрения.

Для достижения поставленной цели необходимо решить следующие задачи:

- Организация и планирование работ над проектом.
- Расчет основных затрат и расходов на проект.
- Оценка экономической эффективности проекта.

4.1 Организация и планирование работы

Одной из составляющих успешной реализации проекта служит рациональное планирование занятости каждого из его участников, а также определение сроков выполнения определенных этапов работы над проектом. В данном разделе приводится перечень этапов работы и исполнителей. Данные по перечню работ и продолжительности работ представлены в таблице 4.1 Число исполнителей данного проекта равно двум – непосредственный исполнитель и научный руководитель.

Таблица 4.1 – Перечень работ и продолжительность их выполнения

Этапы работы	Длительность, дни	Дата начала работ	Дата окончания	Исполнители
1. Постановка задачи, определение цели	1	1.02	1.02	НР
2. Анализ существующих аналогов	5	2.02	8.02	И
3. Исследование и выбор методов детектирования	4	9.02	14.02	НР, И
4. Исследование и выбор методов обработки изображения	4	15.02	20.02	НР, И
5. Выбор программного обеспечения, библиотеки, языка программирования	6	22.02	1.03	НР, И
6. Подготовка обучающей и тестовой выборки	8	2.03	13.03	И
7. Реализация и сопоставление методов детектирования	30	14.03	24.04	И
8. Улучшение выбранного метода	13	25.04	15.05	И
9. Тестирование и анализ результатов	5	16.05	22.05	НР, И
10. Расчет экономических показателей	2	23.05	24.05	И
11. Оценка показателей безопасности жизнедеятельности	2	25.05	26.05	И
12. Оформление пояснительной записки	2	27.05	29.05	И
13. Подведение итогов	2	30.05	31.05	НР, И
Итого:	82	1.02	31.05	
Инженер	81			
Научный руководитель	22			

4.2 Продолжительность этапов работ

Расчет продолжительности этапов работ осуществляется двумя методами:

- технико-экономическим;
- опытно-статистическим.

В данном разделе осуществляется расчет продолжительности работ с

использованием опытно-статистического метода. Данный метод имеет 2 способа реализации - аналоговый и экспертный [40].

- Для определения ожидаемого значения продолжительности работ $t_{ож}$ применяется вероятностный метод – метод двух оценок t_{min} и t_{max} .

Ожидаемые (вероятные) значения продолжительности работ($t_{ож}$):

$$t_{ож} = \frac{3 \cdot t_{min} + 2 \cdot t_{max}}{5}, \quad (4.1)$$

где t_{min} – минимальная трудоемкость работ, чел/дн.; t_{max} – максимальная трудоемкость работ, чел/дн.

- Для выполнения перечисленных в таблице 4.1 работ требуются специалисты:

- инженер – в его роли действует исполнитель НИР (ВКР);
- научный руководитель.

- Для построения линейного графика необходимо рассчитать длительность этапов в рабочих днях, а затем перевести в календарные дни. Расчет продолжительности выполнения каждого этапа ведется по формуле:

$$T_{РД} = \frac{t_{ож}}{K_{ВН}} \cdot K_{Д}, \quad (4.2)$$

где $t_{ож}$ – трудоемкость работы, чел/дн.; $K_{ВН}$ – коэффициент выполнения работ ($K_{ВН} = 1$); $K_{Д}$ – коэффициент, учитывающий дополнительное время на компенсации и согласование работ ($K_{Д} = 1.2$).

- Расчет продолжительности этапа в календарных днях ведется по формуле:

$$T_{КД} = T_{РД} \cdot T_{К}, \quad (4.3)$$

где $T_{РД}$ – продолжительность выполнения этапа в рабочих днях; $T_{КД}$ – продолжительность выполнения этапа в календарных днях; $T_{К}$ – коэффициент календарности, служит для перехода от длительности работ в рабочих днях к их аналогам в календарных днях. Данный коэффициент имеет следующую формулу:

$$T_K = \frac{T_{КАЛ}}{T_{КАЛ} - T_{ВД} - T_{ПД}}, \quad (4.4)$$

где $T_{КАЛ}$ – календарные дни ($T_{КАЛ} = 365$); $T_{ВД}$ – выходные дни (52 или 104 дня, для шестидневной и пятидневной рабочей недели соответственно); $T_{ПД}$ – праздничные дни ($T_{ПД} = 10$).

Таким образом, для шестидневной рабочей недели получаем следующий коэффициент календарности:

$$T_K = \frac{365}{365 - 52 - 10} = 1,205.$$

На рисунке 4.1 представлены задачи исполнителя проекта.

Рк за	Название задачи	Длитель	Начало	Окончание
★	1. Постановка задачи, определение	1 день	Ср 01.02.17	Ср 01.02.17
★	2. Анализ существующих аналогов	5 дней	Чт 02.02.17	Ср 08.02.17
★	3. Исследование и выбор методов де	4 дней	Чт 09.02.17	Вт 14.02.17
★	4. Исследование и выбор методов об	4 дней	Ср 15.02.17	Пн 20.02.17
★	5. Выбор программного обеспечения	6 дней	Ср 22.02.17	Ср 01.03.17
★	6. Подготовка обучающей и тестовой	8 дней	Чт 02.03.17	Пн 13.03.17
★	7. Реализация и сопоставление метод	30 дней	Вт 14.03.17	Пн 24.04.17
★	8. Улучшение выбранного метода	15 дней	Вт 25.04.17	Пн 15.05.17
★	9. Тестирование и анализ результатов	5 дней	Вт 16.05.17	Пн 22.05.17
★	10. Расчет экономических показателей	2 дней	Вт 23.05.17	Ср 24.05.17
★	11. Оценка показателей безопасности	2 дней	Чт 25.05.17	Пт 26.05.17
★	12. Оформление пояснительной записки	2 дней	Сб 27.05.17	Пн 29.05.17
★	13. Подведение итогов	2 дней	Вт 30.05.17	Ср 31.05.17

Рис. 4.1 - Задачи для исполнителя проекта

Диаграмма Ганта для задач исполнителя проекта представлена в приложениях Б.

4.3 Расчет сметы затрат на выполнение проекта

С учетом специфики проделанной работы для рассматриваемого проекта производится оценка следующих расходов:

- основная заработная плата;
- социальный налог;
- расходы на электроэнергию (без освещения);
- прочие расходы.

Ввиду отсутствия привлечения помощи сторонних организаций, командировок, привлечения консалтинговых услуг, расходы по соответствующим статьям не оцениваются применительно к данной работе. Помимо вышеперечисленного, не рассматриваются также затраты на материалы ввиду их незначительности в масштабе данной работы.

4.3.1 Расчет заработной платы

В ТПУ оклады распределены в соответствии с занимаемыми должностями, например, ассистент, ст. преподаватель, доцент, профессор. Базовый оклад Z_6 определяется исходя из размеров окладов, определенных штатным расписанием предприятия. В соответствие с утвержденным Положением об оплате труда в Томском политехническом университете ставки почасовой оплаты для научного руководителя с ученой степенью доцента равна $Z_{\text{дн.науч.р}} = 300$ рублей, для инженера незащищенных специалистов $Z_{\text{дн.инж}} = 175$ рублей.

Основная заработная плата ($Z_{\text{осн}}$) инженера и научного руководителя рассчитывается по следующей формуле:

$$Z_{\text{осн}} = Z_{\text{поч}} * T_p, \quad (4.5)$$

где $Z_{\text{осн}}$ – основная заработная плата работника; T_p – продолжительность работ, выполняемых научно-техническим работником, раб. час; $Z_{\text{поч}}$ – почасовая оплата работника, руб.

При расчете учитывалось, что работа велась в период 1.02- 31.05, всего 82 дней при средней загрузке 2 часа в день. Затраты времени на выполнение работы по каждому исполнителю брались из таблицы 2:

- научный руководитель – 22 дней, всего $T_{р.науч.р} = 44$ часов.
- инженер – 81 дней, всего $T_{р.инж} = 162$ часов.

$$Z_{осн.науч.р} = 300 * 44 = 13\ 200 \text{ рублей,}$$

$$Z_{осн.инж} = 175 * 162 = 28\ 350 \text{ рублей.}$$

Общая сумма заработной платы инженера и научного руководителя составило 41 550 рублей.

4.3.2 Расчет затрат на социальный налог

Ставка взноса в социальные фонды при выполнении научных, опытно-конструкторских и хоздоговорных работ установлен в размере 27,1% от заработной платы. Размер взноса рассчитываются по формуле:

$$C_{соц} = C_{зп} \cdot 0,271\%, \quad (4.6)$$

где $C_{зп}$ – размер заработной платы.

$$C_{соц} = 41\ 550 \cdot 0,271 = 11\ 260,05 \text{ руб.}$$

4.3.3 Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию, используемую оборудованием в ходе выполнения проекта и рассчитывается по формуле:

$$C_{эл.об} = P_{об} \cdot t_{об} \cdot C_э, \quad (4.7)$$

где $P_{об}$ – мощность, потребляемая оборудованием, кВт; $t_{об}$ – время работы оборудования, час; $C_э$ – тариф на 1кВт·час. Значения параметров:

- 1) $C_э$ - для ТПУ составляет 5,8 руб/ кВт·час (с НДС).
- 2) $t_{об} = T_{рД} \cdot K_t$,

где $T_{РД}$ – трудозатраты исполнителя из расчета на 8 часовой рабочий день;
 $K_t \leq 1$ – коэффициент использования оборудования по времени (в данном случае используется значение 0,09).

$$3) P_{об} = P_{ном} \cdot K_c ,$$

где $P_{ном}$ – номинальная мощность оборудования, кВт; $K_c \leq 1$ – коэффициент загрузки, зависит от средней степени использования номинальной мощности. Для данного проекта используется значение 1.

Расчет затрат на электроэнергию представлен в таблице 4.2.

Таблица 4.2 – Затраты на электроэнергию технологическую

Наименование оборудования	Время работы оборудования $t_{об}$, час	Потребляемая мощность $P_{об}$, кВт	Тариф $C_э$, руб/кВт·час	Затраты ЭОБ, руб.
Персональный компьютер инженера	162	0,09	5,8	84,56
Итого				84,56

4.3.4 Расчет прочих расходов

Накладные расходы составляют 40% от всей суммы затрат. Расчет накладных расходов ведется по следующей формуле:

$$C_{нак} = S_{об.сум.зат} * 0,4\% , \quad (4.8)$$

где $S_{об.сум.зат}$ – общая сумма всех затрат.

$$C_{нак} = 52\,894 * 0,4\% = 21\,157,6$$

4.3.5 Расчет общей себестоимости разработки

Общая себестоимость разработки представляет суммарное значение затрат по всем статьям сметы затрат на разработку. Расчет общей себестоимости разработки представлен в таблице 4.3.

Таблица 4.3 – Смета затрат на разработку проекта

Статья затрат	Условное обозначение	Сумма, руб.
Основная заработная плата	Сзп	41 550
Отчисления в социальные фонды	Ссоц	11 260, 05
Расходы на электроэнергию	Сэл.	84,56
Прочие расходы	Спроч	21 157,6
Итого		74 052,21

4.3.6 Расчет прибыли

Одним из вариантов приблизительной оценки прибыли является определение ее размера в пределах от 5 % до 20% от полной себестоимости проекта. Приведем для данной проекта расчет минимальной и максимальной прибыли исходя из заданных пределов. Таким образом:

$$\text{максимальная прибыль} = 0,2 \cdot 74\,052,21 = 14\,810,4 \text{ руб.},$$

$$\text{минимальная прибыль} = 0,05 \cdot 74\,052,21 = 3\,702,6 \text{ руб.}$$

4.4 Оценка экономической эффективности проекта

Выполнение научно-исследовательских работ оценивается уровнями достижения экономического, научного, научно-технического и социального эффектов.

Научный эффект характеризует получение новых научных знаний и отображает прирост информации, предназначенной для внутринаучного потребления. Научно-технический эффект характеризует возможность использования результатов в других проектах и обеспечивает получение информации, необходимой для создания новой техники. Экономический эффект характеризуется в частности выраженной в стоимостных показателях экономией живого общественного труда, а также в возможности применения полученных знаний для создания новых разработок. Социальный эффект проявляется в улучшении условий труда [41].

Экономическая эффективность проекта обусловлена возрастающей необходимостью в системах автоматизации детектирования автомобильных номеров. Конечная система детектирования автомобильных номеров является востребованной и актуальной на сегодняшний день.

Подводя итог вышесказанному, экономический эффект от реализации проекта может быть выражен в снижении затрат на покупку подобных систем от сторонних разработчиков, а так же повышение эффективности детектирования номерных знаков за счет экономии времени при замене ручной обработки на автоматизированную.

4.5 Оценка научно-технического уровня НИР

Научно-технический уровень характеризует влияние проекта на уровень и динамику обеспечения научно-технического прогресса в данной области. Для данной оценки используется метод балльных оценок, сущность которого заключается в присвоении каждому из показателей НИР определенного количества баллов по соответствующей для данного показателя шкале.

Научно-технический уровень определяется на основании его интегрального показателя, который выражается следующей формулой:

$$I_{НТУ} = \sum_1^3 R_i \cdot n_i, \quad (4.9)$$

где $I_{НТУ}$ – интегральный индекс научно-технического уровня; R_i – весовой коэффициент i -го признака научно-технического эффекта; n_i – количественная оценка i -го признака научно-технического эффекта в баллах;

Оценка научно-технического уровня представлена в таблице 4.4.

Таблица 4.4 – Оценка научно технического уровня НИР

Значимость	Фактор НТУ	Уровень фактора	Выбранный балл	Обоснование выбранного балла
0,4	Уровень новизны	Принципиально новая	8	Новый способ организации информационной инфраструктуры, новый стандарт передачи данных
0,1	Теоритический уровень	Разработка способа	6	Разработка нового способа объединения информации
0,5	Возможность реализации	В течение первых лет	10	Реализуется на основе хорошо известных технологий

Таким образом, для данного проекта показатель НТУ равен:

$$I_{\text{НТУ}} = 0,4 \cdot 8 + 0,1 \cdot 6 + 0,5 \cdot 10 = 3,2 + 0,6 + 5 = 8,8.$$

В таблице 4.5 приводится оценка качественных уровней НИР.

Таблица 4.5 – Качественная оценка показателей НИР

Уровень НТЭ	Низкий	Средний	Высокий
Показатель НТЭ	1-4	4-7	8-10

На основании таблицы 4.5 данная работа относится к высокому уровню научно технического эффекта. Высокий уровень НТЭ обусловлен тем, что данная работа представляет принципиально новый подход в организации информационной метеорологической станции.

4.6 Выводы по Главе 4

В данной главе была произведена оценка различных экономических аспектов разработанного проекта. На основании полученных результатов проекту можно дать следующую характеристику:

- по времени разработки проект относится к краткосрочному виду проектов;
- по масштабности проект можно отнести к категории малых проектов.

5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

Целью магистерской диссертации является разработка и программная реализация алгоритма детектирования номерных знаков на изображениях. Алгоритм детектирования номерных знаков может применяться в правоохранительных органах, в системах контроля автомобилей и транспортных средств.

Для выполнения поставленной задачи проводились теоретические исследования и программная реализация алгоритма проводилось с использованием персонального компьютера.

Использование средств вычислительной техники, влияет на производительность труда работника и может привести к существенным проблемам со здоровьем.

По этой причине целью данного приложения является исследование оценки условий труда на рабочем месте, анализ влияния вредных и опасных факторов труда, организационные мероприятия по предотвращению ЧС, по защите окружающей среды. А также правовые и организационные вопросы обеспечения безопасности.

5.1 Профессиональная и социальная безопасность

5.1.1 Анализ выделенных вредных и опасных факторов, которые могут возникнуть при разработке проектируемого решения

Работа связана с анализом и теоретической разработкой алгоритма детектирования номерных пластин на изображениях, и его реализацией. В ходе выполнения работы могут возникнуть следующие вредные и опасные факторы:

- 1) нарушение параметров микроклимата;
- 2) недостаточная освещенность рабочей зоны;
- 3) монотонный режим работы;
- 4) опасность поражения электрическим током;
- 5) опасность возникновения пожара.

Для обеспечения безопасности в рабочей зоне необходимо проанализировать влияние вредных и опасных факторов, а также возникновение чрезвычайных ситуаций [39].

5.1.1.1 Нарушение параметров микроклимата

Присутствие вредных факторов на рабочем месте неблагоприятно сказывается на здоровье человека. Для предотвращения возникновения неблагоприятных условий труда необходимо учитывать следующие возможные влияния вредных факторов.

Повышенная температура поверхностей ПК и воздуха в рабочей зоне оказывает негативное влияние на нервную систему человека, психологическая атмосфера в помещении может дестабилизироваться и стать причиной возникновения конфликтов в коллективе. Высокая температура может послужить причиной роста болезнетворных бактерий, которые влекут за собой риск возникновения разного рода заболеваний.

Влажность в помещении играет немаловажную роль в продуктивной работе сотрудников. При пониженной влажности воздуха происходит осушение слизистых оболочек человека. Такие условия могут привести к повышенному накоплению вредных бактерий и вирусов в дыхательных путях. Также неблагоприятными условиями является повышенная влажность в офисном помещении, которая способствует размножению грибков, плесени, бактерий. Повышенное содержание таких микроорганизмов способствует возникновению у человека аллергических и хронических заболеваний дыхательных путей.

В санитарных нормах установлены величины параметров микроклимата, создающие комфортные условия. Нормы устанавливаются в зависимости от времени года, характера трудового процесса и характера производственного помещения. В таблица 5.1 приведены оптимальные нормы параметров микроклимата.

Таблица 5.1 – Оптимальные нормы параметров микроклимата

Период года	Параметр микроклимата	Величина
Холодный	Температура воздуха в помещении	22...24 °С
	Относительная влажность	40...60 %
	Скорость движения воздуха	до 0,1 м/с
Теплый	Температура воздуха в помещении	23...25 °С
	Относительная влажность	40...60 %
	Скорость движения воздуха	0,1...0,2 м/с

Согласно требованиям СанПиН 2.2.2/2.4.1340–03, в кабинете поддерживается температура равная 19–20 С°, при относительной влажности в 55–58%. Для этого в кабинете проводится ежедневная влажная уборка и систематическое проветривание после каждого часа работы на ПЭВМ [43] [44].

5.1.1.2 Недостаточная освещенность рабочей зоны

Естественное освещение создается прямыми солнечными лучами или рассеянным светом небосвода. Его следует предусматривать для всех производственных, складских, санитарно-бытовых и административных помещений.

Спектр естественного освещения наиболее благоприятен для глаз человека. Входящее в состав солнечного спектра ультрафиолетовое излучение имеет важное значение для здоровья человека, однако оно практически полностью задерживается при прохождении сквозь обычное стекло, поэтому не проникает внутрь помещений.

Различают следующие виды производственного освещения: естественное, искусственное и совмещенное. Естественное освещение осуществляется за счет прямого и отраженного света неба. С физиологической точки зрения естественное освещение наиболее благоприятно для человека. Естественное освещение в течение дня меняется в достаточно широких пределах в зависимости от состояния атмосферы (облачность). Различают боковое естественное освещение — через световые проемы (окна) в наружных стенах и верхнее естественное

освещение, при котором световой поток поступает через световые проемы, расположенные в верхней части (крыше) здания (аэрационные и зенитные фонари и т.д.). Если используется оба вида освещения, то оно называется комбинированным. Для характеристики естественного освещения используется коэффициент естественной освещенности (КЕО). Величины КЕО для различных помещений лежат в пределах 0,1–12%. Уровень естественной освещенности в производственных помещениях с течением времени снижается вследствие загрязнения остекленных поверхностей, стен и потолков. Поэтому следует регулярно чистить стекла, красить или белить стены и потолки. Такие мероприятия необходимо выполнять тем чаще, чем выше концентрация пыли или других взвешенных в воздухе веществ.

Слепящее действие прямых солнечных лучей на работающих и возникающую при этом блёскость предметов устраняют с помощью солнцезащитных козырьков, штор, жалюзи и экранов.

Освещение играет немаловажную роль в рабочем процессе. Часто в офисных помещениях недостаточно естественного света. В таком случае необходимо возмещать его недостаток искусственным освещением.

При нехватке освещенности человек испытывает дискомфорт и вследствие всего могут развиваться психические заболевания, ухудшение зрения, снижение концентрации. В рассматриваемом рабочем месте освещенность на поверхности стола в зоне размещения рабочего документа равна 300 лк. Освещение не создает бликов на поверхности экрана. Освещенность поверхности экрана равна примерно 300 лк.

Показатели соответствуют нормам СанПиН 2.2.2/2.4.1340–03 [45].

Разряд зрительных работ программиста и оператора ПЭВМ относится к разряду III и подразряду г (работы высокой точности). В таблице 5.2 представлены нормативные показатели искусственного освещения при работах заданной точности.

Таблица 5.2 – Требования к освещению помещений промышленных предприятий для операторов ПЭВМ [49]

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта различения, мм	Разряд зрительной работы	Подразряд зрительной работы	Контраст объекта с фоном	Характеристика фона	Искусственное освещение		
						Освещённость, лк		
						При системе комбинированного освещения		При системе общего освещения
всего	в том числе от общего							
Высокой точности	От 0,3 до 0,5	III	г	Средний, большой	Светлый, средний	400	200	200

5.1.1.3 Монотонный режим работы

Большое количество информации, воспринимаемая программистом, при работе с ПЭВМ воздействует неблагоприятно на нервную систему программиста или пользователя. Это является непростой задачей, которая очень сильно воздействует на сознание и психофизическое состояние из-за монотонности работы. Поэтому меры, позволяющие снизить воздействие этого вредного производственного фактора, которые регулируются СанПиН 2.2.2/2.4.1340-03, являются важными в работе оператора ПЭВМ. Они позволяют увеличить производительность труда и предотвратить возникновения профессиональных болезней.

Организация работы с ПЭВМ осуществляется в зависимости от вида и категории трудовой деятельности. Виды трудовой деятельности разделяются на 3 группы: группа А – работа по считыванию информации с экрана с предварительным запросом; группа Б – работа по вводу информации; группа В – творческая работа в режиме диалога с ПЭВМ. Работа программиста-разработчика рассматриваемой в данной работе системы относится к группам А и Б, в то время,

как деятельность специалиста, который будет использовать систему в профессиональной деятельности, относится к группе В.

Категории туровой деятельности различаются по степени тяжести выполняемых работ. Для снижения воздействия рассматриваемого вредного фактора предусмотрены регламентированные перерывы для каждой группы работ – таблица 5.3.

Таблица 5.3 – Суммарное время регламентированных перерывов в зависимости от продолжительности работы, вида категории трудовой деятельности с ПЭВМ [45].

Категория работы с ПЭВМ	Уровень нагрузки за рабочую смену при видах работ с ПЭВМ			Суммарное время регламентированных перерывов, мин.	
	группа А, количество знаков	группа Б, количество знаков	группа В, ч	при 8-часовой смене	при 12-часовой смене
I	до 20 000	до 15 000	до 2	50	80
II	до 40 000	до 30 000	до 4	70	110
III	до 60 000	до 40 000	до 6	90	140

5.1.1.4 Опасность поражения электрическим током

Поражение электрическим током является опасным производственным фактором и, поскольку оператор ПЭВМ имеет дело с электрооборудованием, то вопросам электробезопасности на его рабочем месте должно уделяться много внимания. Нормы электробезопасности на рабочем месте регламентируются СанПиН 2.2.2/2.4.1340-03, вопросы требований к защите от поражения электрическим током освещены в ГОСТ Р 12.1.019-2009 ССБТ.

Электробезопасность – система организационных и технических мероприятий и средств, обеспечивающих защиту людей от вредного и опасного воздействия электрического тока, электрической дуги, электромагнитного поля и статического электричества.

Опасность поражения электрическим током усугубляется тем, что человек не в состоянии без специально предназначенных приборов обнаружить напряжение дистанционно.

Помещение, где находится рабочее место оператора ПЭВМ, относится к помещениям без повышенной опасности ввиду отсутствия следующих факторов: влажность, токопроводящая пыль, токопроводящие полы, высокая температура, возможность одновременного прикосновения человека к имеющим соединение с землей металлоконструкциям зданий, технологическим аппаратам, механизмам и металлическим корпусам электрооборудования.

При работе с электрическим оборудованием пользователь ПЭВМ обязательно должен соблюдать следующие меры предосторожности:

- Перед началом работы нужно удостовериться, что выключатели и розетка закреплены и не имеют оголённых токоведущих частей;
- При обнаружении поврежденности оборудования и приборов необходимо, не делая никаких самостоятельных исправлений, сообщить человеку, ответственному за оборудование [46, 47].

5.1.1.5 Опасность возникновения пожара

Пожары приводят к полной потере информации, большим трудностям восстановления всей информации в полном объёме.

В помещениях с ПЭВМ повышен риск появления пожара из-за наличия множества факторов: наличие значительного количества электронных схем, устройств электропитания, устройств кондиционирования воздуха; возможные неисправности электрооборудования, света, или неправильная их эксплуатация может послужить причиной пожара.

Основой воспламенения могут быть: неисправности электропроводки, розеток и выключателей которые могут привести к короткому замыканию или пробоем изоляции; использование поврежденных (неисправных) электроприборов; использование в помещении электронагревательных приборов с открытыми нагревательными элементами; возникновение пожара, вследствие попада-

ния молнии в здание; неаккуратное обращение с огнем и несоблюдение мер пожарной безопасности [48].

5.1.2 Рекомендации по защите исследователя от действия опасных и вредных факторов

К мероприятиям по оздоровлению воздушной среды в производственном помещении относятся правильная организация вентиляции и кондиционирования воздуха, отопление помещений. Вентиляция может осуществляться естественным и механическим путём. В зимнее время в помещении необходимо предусмотреть систему отопления. Объем помещений с ЭВМ не должен быть меньше $20 \text{ м}^3/\text{человека}$ [43].

Для обеспечения требуемого уровня освещения в помещении используются лампы дневного освещения, равномерно распределенные по всему потолку офиса. Для освещения помещения выбраны наиболее широко применяемые лампы типа ЛБ.

Для обеспечения комфортных условий используются как организационные методы (рациональная организация проведения работ в зависимости от времени года и суток, чередование труда и отдыха), так и технические средства (вентиляция, кондиционирование воздуха, отопительная система).

При возникновении несчастного случая следует немедленно освободить пострадавшего от действия электрического тока и, вызвав врача, оказать ему необходимую помощь. Для защиты от поражения электрическим током все токоведущие части должны быть защищены от случайных прикосновений кожухами, корпус устройства должен быть заземлен. Заземление выполняется изолированным медным проводом сечением $1,5 \text{ мм}^2$, который присоединяется к общей шине заземления с общим сечением $5,4 \text{ мм}^2$ при помощи сварки. Общая шина присоединяется к заземлению, сопротивление которого не должно превышать 4 Ом . Питание устройства должно осуществляться от силового щита

через автоматический предохранитель, который срабатывает при коротком замыкании нагрузки.

Для устранения возможности пожара в помещении должны соблюдаться следующие противопожарные меры:

- ограничение количества горючих веществ;
 - устранение возможных источников возгорания (электрических искр, нагрева оболочек оборудования);
 - применение средств пожаротушения;
 - использование пожарной сигнализации;
 - содержание электрооборудования в исправном состоянии, использование плавких предохранителей и автоматических выключателей в аппаратуре, по окончании работ все установки должны обесточиваться;
 - наличие в помещении средств пожаротушения (огнетушители типа ОУ-3, пожарный инструмент, песок) и содержание их в исправном состоянии;
 - разрешение курения в только отведенных для этого местах;
 - содержание путей и проходов эвакуации людей в свободном состоянии;
 - проводить раз в год инструктаж по пожарной безопасности;
- назначение ответственного за пожарную безопасность помещения.

5.2 Экологическая безопасность

5.2.1 Анализ влияния процесса разработки на окружающую среду

В данном разделе рассматривается влияние процесса исследования на окружающую среду, а также самого продукта в результате его реализации на производстве.

Программный продукт, разработанный в ходе выполнения ВКР, не наносит вреда окружающей среде ни на стадиях его разработки, ни на стадиях эксплуатации. Однако, средства, необходимые для его разработки и эксплуатации могут наносить вред окружающей среде.

Современные ПЭВМ производят практически без использования вредных веществ, опасных для человека и окружающей среды. Исключением являются аккумуляторные батареи компьютеров и мобильных устройств. В аккумуляторах содержатся тяжелые металлы, кислоты и щелочи, которые могут нанести ущерб окружающей среде, попадая в гидросферу и литосферу, если они были неправильно утилизированы. Для утилизации аккумуляторов необходимо обращаться в специальные организации, специализировано занимающиеся приемом, утилизацией и переработкой аккумуляторных батарей [49].

Люминесцентные лампы, применяющиеся для искусственного освещения рабочих мест, также требуют особой утилизации, т.к. в них содержится от 10 до 70 мг ртути, которая относится к чрезвычайно-опасным химическим веществам и может стать причиной отравления живых существ, а также загрязнения атмосферы, гидросферы и литосферы. Сроки службы таких ламп составляют около 5-ти лет, после чего их необходимо сдавать на переработку в специальных пунктах приема. Юридические лица обязаны сдавать лампы на переработку и вести паспорт для данного вида отходов [49-50].

5.2.2 Обоснование мероприятий по защите окружающей среды

Бумажные отходы должны передаваться в соответствующие организации для дальнейшей переработки во вторичные бумажные изделия.

Неисправные комплектующие персональных компьютеров должны передаваться либо государственным организациям, осуществляющим вывоз и уничтожение бытовых и производственных отходов, либо организациям, занимающимся переработкой отходов. Важнейшим этапам обращения с отходами является их сбор, а в дальнейшем переработка, утилизация и захоронение [55].

В данный момент программный продукт в плане обнаружения детектирует номерную пластину, являющийся прямоугольной областью. В дальнейшем, если доработать программный продукт, то можно использовать в экологии в плане нахождения вредных элементов: контейнеры с вредными

веществами, нефте-газовые контейнеры, атомные радиоактивные контейнеры и биологические отходы больницы находящиеся в неполюженном месте.

5.3 Безопасность в чрезвычайных ситуациях

5.3.1 Перечень возможных ЧС, которые могут возникнуть при разработке

Чрезвычайная ситуация (ЧС) - это обстановка на определенной территории или акватории, сложившаяся в результате аварии, опасного природного явления, катастрофы, стихийного или иного бедствия, которая может повлечь за собой человеческие жертвы, ущерб здоровью людей или окружающей природной среде, значительные материальные потери и нарушение условий жизнедеятельности людей. Различают чрезвычайные ситуации по характеру источника: природные, техногенные, биолого-социальные и военные. А так же по масштабам: локальные, местные, территориальные, региональные, федеральные и трансграничные (из ГОСТ Р 22.0.02-94 с изменением 2000 г «Безопасность в чрезвычайных ситуациях»).

Источник чрезвычайных ситуаций - это опасное природное явление, авария или опасное техногенное происшествие, широко распространенные инфекционные болезни людей, сельскохозяйственных животных и растений, а также применение современных средств поражения, в результате чего происходит ЧС. Источник опасности - любая деятельность или состояние среды, способные привести к реализации опасности или возникновению факторов опасности. По происхождению источники опасности бывают естественными и антропогенными [54].

В рабочей среде оператора ПЭВМ возможно возникновение следующих чрезвычайных ситуаций техногенного характера:

- Пожары и взрывы в зданиях и на коммуникациях;
- Кибертерроризм.

Разработанный программный продукт можно применить для борьбы с терроризмом. Программы такого рода служат для отслеживания движущихся объектов и для распознавания номеров нарушителей закона. Можно применить для проведения следственных действий, с целью расследования преступлений и представления в суд наглядного доказательственного материала.

5.3.2 Меры по ликвидации ЧС и последствий

Помещение для работы операторов ПЭВМ по системе классификации категорий помещений по взрывопожарной и пожарной опасности относится к категории Д (из 5-ти категорий А, Б, В1-В4, Г, Д), т.к. относится к помещениям с негорючими веществами и материалами в холодном состоянии [51].

Каждый сотрудник организации должен быть ознакомлен с инструкцией по пожарной безопасности, пройти инструктаж по технике безопасности и строго соблюдать его.

Запрещается использовать электроприборы в условиях, не соответствующих требованиям инструкций изготовителей, или имеющие неисправности, которые в соответствии с инструкцией по эксплуатации могут привести к пожару, а также эксплуатировать электропровода и кабели с поврежденной или потерявшей защитные свойства изоляцией. Электроустановки и бытовые электроприборы в помещениях по окончании рабочего времени должны быть обесточены (вилки должны быть вынуты из розеток). Под напряжением должны оставаться дежурное освещение и пожарная сигнализация. Недопустимо хранение легковоспламеняющихся, горючих и взрывчатых веществ, использование открытого огня в помещениях офиса.

Перед уходом из служебного помещения работник обязан провести его осмотр, закрыть окна, и убедиться в том, что в помещении отсутствуют источники возможного возгорания, все электроприборы отключены и выключено освещение. С периодичностью не реже одного раза в три года необходимо про-

водить замеры сопротивления изоляции токоведущих частей силового и осветительного оборудования.

Повышение устойчивости достигается за счет проведения соответствующих организационно-технических мероприятий, подготовки персонала к работе в ЧС [49].

Работник при обнаружении пожара или признаков горения (задымление, запах гари, повышение температуры и т.п.) должен:

- Немедленно прекратить работу и вызвать пожарную охрану по телефону «01», сообщив при этом адрес, место возникновения пожара и свою фамилию;
- Принять по возможности меры по эвакуации людей и материальных ценностей;
- Отключить от сети закрепленное за ним электрооборудование;
- Приступить к тушению пожара имеющимися средствами пожаротушения;
- Сообщить непосредственному или вышестоящему начальнику и оповестить окружающих сотрудников;
- При общем сигнале опасности покинуть здание согласно «Плану эвакуации людей при пожаре и других ЧС».

Для тушения пожара применять ручные углекислотные огнетушители (типа ОУ-2, ОУ-5), находящиеся в помещениях офиса, и пожарный кран внутреннего противопожарного водопровода. Они предназначены для тушения начальных возгораний различных веществ и материалов, за исключением веществ, горение которых происходит без доступа воздуха. Огнетушители должны постоянно содержаться в исправном состоянии и быть готовыми к действию. Категорически запрещается тушить возгорания в помещениях офиса при помощи химических пенных огнетушителей (типа ОХП-10) [51].

5.4 Правовые и организационные вопросы обеспечения безопасности

5.4.1 Специальные правовые нормы трудового законодательства

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

Продолжительность рабочего дня не должна быть меньше указанного времени в договоре, но не больше 40 часов в неделю. Для работников до 16 лет – не более 24 часов в неделю, от 16 до 18 лет и инвалидов I и II группы – не более 35 часов.

Возможно установление неполного рабочего дня для беременной женщины; одного из родителей (опекуна, попечителя), имеющего ребенка в возрасте до четырнадцати лет (ребенка-инвалида в возрасте до восемнадцати лет). Оплата труда при этом производится пропорционально отработанному времени, без ограничений оплачиваемого отпуска, исчисления трудового стажа и других прав.

При работе в ночное время продолжительность рабочей смены сокращается на один час. К работе в ночную смену не допускаются беременные женщины; работники, не достигшие возраста 18 лет; женщины, имеющие детей в возрасте до трех лет, инвалиды, работники, имеющие детей-инвалидов, а также работники, осуществляющие уход за больными членами их семей в соответствии с медицинским заключением, матери и отцы-одиночки детей до пяти лет.

Организация обязана предоставлять ежегодный отпуск продолжительностью 28 календарных дней. Дополнительные отпуска предоставляются работникам, занятым на работах с вредными или опасными условиями труда, работникам имеющими особый характер работы, работникам с ненормированным рабочим днем и работающим в условиях Крайнего Севера и приравненных к нему местностях.

В течение рабочего дня работнику должен быть предоставлен перерыв для отдыха и питания продолжительностью не более двух часов и не менее 30 минут, который в рабочее время не включается. Всем работникам предоставляются выходные дни, работа в выходные дни осуществляется только с письменного согласия работника.

Организация-работодатель выплачивает заработную плату работникам. Возможно удержание заработной платы только в случаях установленных ТК РФ ст. 137. В случае задержки заработной платы более чем на 15 дней, работник имеет право приостановить работу, письменно уведомив работодателя. Законодательством РФ запрещена дискриминация по любым признакам и принудительный труд [53].

5.4.2 Организационные мероприятия при компоновке рабочей зоны

К мероприятиям, относящимся к компоновке рабочей зоны относятся работы по организации рабочего места пользователя, позволяющие наилучшим образом организовать деятельность работника, делая его работу максимально удобной и безопасной.

Основным направлением реализованного алгоритма является применение его в качестве программной системы для детектирования номерных знаков в системах контроля автомобилей и транспортных средств. Сам продукт не влияет на организацию рабочей зоны, но работа с ним позволит реорганизовать работу специалистов, что в свою очередь повлияет на организацию рабочей зоны. Это может быть охарактеризовано с помощью следующих факторов:

- позволяет экономить время и усилия, затрачиваемые на ввод данных вручную.
- преимущества автоматического детектирования над визуальным контролем.
- высокая точность и надежность детектирования минимизируют количество ошибок в данных.

Все перечисленные факторы облегчают работу и положительно сказываются на производительности труда.

ЗАКЛЮЧЕНИЕ

Основной целью работы являлась разработка и программная реализация алгоритма детектирования номерных знаков на изображениях. В результате работы над выпускной квалификационной работы был разработан и реализован алгоритм детектирования номерных знаков автомобилей.

В ходе работы были изучены методы поиска объектов на изображениях, были исследованы два базовых метода детектирования: метод контурного анализа и метод Виолы – Джонса. В результате проведенных экспериментов был выбран метод Виолы-Джонса. Основным недостатком данного метода является наличие множественных срабатываний. Для снижения количества ложных срабатываний, а также для увеличения точности детектирования была предложена модификация метода Виолы – Джонса.

В качестве модификации были реализованы различные сглаживающие фильтры для предварительной обработки изображений. В качестве сглаживающих фильтров сравнивались: однородный, фильтр Гаусса, медианный и билатеральный фильтр. Результаты численных экспериментов показали преимущество применения билатерального фильтра. Для снижения количества множественных срабатываний были разработаны следующие функции для вычисления текстурных особенностей: средняя яркость, дисперсия, параметр относительной гладкости R , третий момент, однородность и энтропия. В ходе экспериментов выяснилось, что наиболее стабильным признаком текстурных особенностей номерного знака является энтропия. Было проведено сопоставление с тремя другими алгоритмами детектирования номерных знаков, в результате которого разрабатываемый алгоритм и два алгоритма уступили третьему алгоритму менее 1%.

Была проведена оценка и анализ экономических аспектов и исследовано влияние данной работы на окружающую среду.

Реализованный алгоритм детектирования номерных знаков был протестирован на различных группах изображений и в результате удалось достичь точности детектирования номерных знаков 93,33%.

ОПУБЛИКОВАННЫЕ РАБОТЫ

1. Сыздыкова Г.Ж. Детектирование автомобильных номеров [Электронный ресурс] / Г. Ж. Сыздыкова, науч. рук. Ю. А. Болотова // Молодежь и современные информационные технологии: сборник трудов XIV Международной научно-практической конференции студентов, аспирантов и молодых ученых, г. Томск, 7-11 октября 2016 г. в 2 т. / Национальный исследовательский Томский политехнический университет (ТПУ), Институт кибернетики (ИК) ; ред. кол. В. С. Аврамчук [и др.]. — 2017. — Т. 2. — [С. 195-96]. — Заглавие с титульного экрана. — Свободный доступ из сети Интернет. — Adobe Reader.

Режим доступа: [http://portal.tpu.ru:7777/f_ic/files/science/activities/msit/msit2016/Sbornik_2016/Sbornik_MSIT_2016_\(Tom2\).pdf](http://portal.tpu.ru:7777/f_ic/files/science/activities/msit/msit2016/Sbornik_2016/Sbornik_MSIT_2016_(Tom2).pdf)

2. Сыздыкова Г.Ж. Исследование влияния алгоритмов фильтрации на улучшение качества детектирования номерных пластин методом Виолы - Джонса / Г. Ж. Сыздыкова, науч. рук. Ю. А. Болотова // Научная сессия ТУСУР–2017: материалы Международной научно-технической конференции студентов, аспирантов и молодых ученых, посвященной 55-летию ТУСУРа, Томск, 10–12 мая 2017 г.: в 8 частях. – Томск: В-Спектр, 2017 – Ч. 5. – [С. 99-101].

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Компьютерное зрение [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 08.03.17 г.
2. Sobel I., Feldman G. A 3x3 isotropic gradient operator for image processing, presented at a talk at the Stanford Artificial Project, in Pattern Classification and Scene Analysis, R. Duda and P. Hart, Eds., // John Wiley & Sons, – 1968, – P. 271–272.
3. Robinson G. Edge detection by compass gradient masks, Computer Graphics Image Precossesing 6, 1977, 492-501.
4. Viola P., Jones M. J. Robust real-time face detection // International Journal of Computer Vision. – 2004. – V. 57. – P. 137-154.
5. Linda G. Shapiro and George C. Stockman. Computer Vision // New Jersey, Prentice-Hall, –2001. – P. 279-325.
6. Фурман, Я. А. Цифровые методы обработки и распознавания бинарных изображений / Я. А. Фурман, А. Н. Юрьев, В. В. Яншин. – Красноярск: Изд-во Краснояр. ун-та, 1992.— 248 с.
7. OpenCV шаг за шагом. Обработка изображения - детектор границ Канни [Электронный ресурс]. URL: <http://robocraft.ru/blog/computervision>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 9.09.2016 г.
8. Вершинина В.В., Паламарь И.Н. Организация базы знаний семантическойсети на основе XML-формата. Тез. докл. IV ВНТК «Информационные технологии в науке, проектировании и производстве». Нижний Новгород, МВВО АТН РФ, 2002, с. 23.
9. Оператор Кэнни [Электронный ресурс]. URL: <http://ru.wikipedia.org/wiki/>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 9.09.2016 г.
10. Алгоритм Рамера—Дугласа—Пекера [Электронный ресурс]. URL: <http://ru.wikipedia.org/wiki/>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 9.09.2016 г.

- 11.Метод Виолы – Джонса [Электронный ресурс]. URL: http://ru.wikipedia.org/wiki/Метод_Виолы-Джонса. Дата обращения:10.09.2016 г.
- 12.P. Viola и J. Michael, «Rapid Object Detection using a Boosted Cascade of Simple» в Computer Vision and Pattern Recognition, 2001.
- 13.Метод Виолы-Джонса как основа для распознавания лиц [Электронный ресурс]. URL: <http://habrahabr.ru/post/133826/>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 5.09.2016 г.
14. Lienhart R., Kuranov E., Pisarevsky V.: Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In: PRS 2003, pp. 297-304, 2003.
- 15.Hrvoje Rudeš, Petra Grd, License plate detection for preserving privacy using Haar classifiers// Central European Conference on Information and Intelligent Systems. September 23-25, 2015. Pp 153- 157.
- 16.Cascade Classifier Training [Электронный ресурс]. URL: http://docs.opencv.org/2.4.10/doc/user_guide/ug_traincascade/, свободный. – Загл. с экрана. – Яз. англ. Дата обращения: 10.09.2016 г.
- 17.Лекция Интернет Университета «Фильтрация изображений» [Электронный ресурс]. URL: <http://www.intuit.ru/>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 01.12.16 г.
- 18.Гонзалес, Р. Цифровая обработка изображений /Р. Гонзалес, Р. Вудс. – М.: Техносфера, 2005. – 1072с.
- 19.Грузман И.С., Киричук В.С., Косых В.П., Перетягин Г.И., Спектор А.А. Цифровая обработка изображений в информационных системах: Учеб. пособие. – Новосибирск.: Изд-во НГТУ, 2003. – 352 с.
- 20.Сато Ю. Обработка сигналов. Первое знакомство. 2-издание. – М.: Додэка XXI, 2009. –176с.
- 21.Оппенгейм А. Шафер Р. Цифровая обработка сигналов: 2-издание. – М.: Техносфера.

22. Лайонс Ричард. Цифровая обработка сигналов: 2 изд. – М.: ООО Бином-Пресс, 2006. – 656 с.
23. Сергиенко А.Б. Цифровая обработка сигналов. – СПб.: Питер, 2007. -752 с.
24. Фисенко В.Т., Фисенко Т.Ю., Компьютерная обработка и распознавание изображений: учеб. пособие. – СПб: СПбГУ ИТМО, 2008.–192 с.
25. Яне Б. Цифровая обработка изображений. – М.: Техносфера, 2007. – 584 с.
26. Шапиро Л., Стокман Дж. Компьютерное зрение. – М.: БИНОМ. Лаборатория знаний, 2006. – 752 с.
27. Базовые операции обработки изображений [Электронный ресурс]. URL: <http://www.intuit.ru/studies/>, свободный. – Загл. с экрана. – Яз. рус.. Дата обращения: 9.02.2017 г.
28. Tomasi C., Manduchi R.: Bilateral filtering for gray and color images// Proc.of International Conference on Computer Vision, IEEE, 1998. –P. 839–846.
29. Крапчатова Т. В., Филиппов М. В. Анализ эффективности алгоритмов билатеральной фильтрации. Издатель ФГБОУ ВПО "МГТУ им. Н.Э. Баумана". Эл № ФС 77 - 48211. ISSN 1994-0408.
30. Выравнивание гистограмм [Электронный ресурс]. URL: <http://www.intuit.ru/studies/>, свободный. – Загл. с экрана. – Яз. рус.. Дата обращения: 9.02.2017 г.
31. А. К. Jain, "Fundamentals of Digital Image Processing Englewood Cliffs," NJ: Prentice Hall, 1989.
32. Яковлев А. Методы анализа и синтеза текстур [Электронный ресурс]. URL: science/paper/paper1.pdf. Дата обращения: 9.02.2017 г.
33. Болотова Ю.А., Друки А.А., Спицын В.Г. Алгоритмы интеллектуальной обработки цифровых изображений: учебное пособие / Ю.А. Болотова, А.А. Друки, В.Г. Спицын; Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2016. – 217 с.
34. Вежнев А. Методы сегментации изображений: автоматическая

- сегментация / А. Вежнев, О. Барина // Сетевой журнал Компьютерная графика и мультимедия. 2006. [Электронный ресурс]. – URL: <http://cgm.computergraphics.ru/content/view/147>. Дата обращения: 15.03.2017 г.
35. Язык программирования [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/>, свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 9.03.2017 г.
36. Bradski G., Kaehler A.: Learning OpenCV. O'Reilly Media, Inc, 2008. – 571с.
37. Rasheed S., Naeem A., Ishaq O.: Automated Number Plate Recognition Using Hough Lines and Template Matching // Proceedings of the World Congress on Engineering and Computer Science 2012 Vol I WCECS 2012. In: San Francisco, USA, October 24-26, 2012.
38. Hung K., Hsieh C.: A Real-Time Mobile Vehicle License Plate Detection and Recognition // Tamkang Journal of Science and Engineering, Vol. 13, No. 4, pp. 433-442, 2010.
39. Hermawati F., Budianto H.: A Video Based License Plate Detection System Using Viola-Jones Method // Department of Informatics, Faculty of Engineering, University of Surabaya.
40. Гаврикова Н.А. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение: учебно-методическое пособие / Н.А. Гаврикова, Л.Р. Тухватулина, И.Г. Видяев, Г.Н. Серикова, Н.В. Шаповалова; Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2014. – 73 с.
41. Д.В. Арутюнова. Инновационный менеджмент: учебное пособие / Ростов-на-Дону: Изд-во ЮФУ, 2014. – 152 с.
42. ГОСТ 12.0.003-74. ССБТ. Опасные и вредные производственные факторы. Классификация.
43. СанПиН 2.2.2/2.4.1340–03. Санитарно-эпидемиологические правила и нормативы «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

44. СанПиН 2.2.4.1191–03. Электромагнитные поля в производственных условиях.
45. СанПиН 2.2.1/2.1.1.1278–03. Гигиенические требования к естественному, искусственному и совмещённому освещению жилых и общественных зданий.
46. ГОСТ Р 17.0.0.06-2000. Охрана природы. Экологический паспорт природопользователя. Основные положения. Типовые формы.
47. ГОСТ Р 22.0.01-94. Безопасность в ЧС. Основные положения.
48. ГОСТ Р 22.0.07-95. Безопасность в чрезвычайных ситуациях. Источники техногенных чрезвычайных ситуаций. Классификация и номенклатура поражающих факторов и их параметров.
49. Федеральный закон от 22.07.2013 г. №123 – ФЗ. Технический регламент о требованиях пожарной безопасности.
50. Трудовой кодекс Российской Федерации (ТК РФ). ГОСТ 12.2.032-78. Система стандартов безопасности труда. Рабочее место при выполнении работ сидя. Общие эргономические требования.
51. НПБ 105-03 Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности // Электронный фонд правовой и нормативно-технической документации. 2016. URL: <http://docs.cntd.ru/document/1200032102>. Дата обращения: 24.04.2017 г.
52. ППБ 01–03. Правила пожарной безопасности в Российской Федерации. – М.: Министерство Российской Федерации по делам гражданской обороны, чрезвычайным ситуациям и ликвидации последствий стихийных бедствий, 2003.
53. Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред.от 24.04.2017) // Консультант Плюс, 2015.URL: http://www.consultant.ru/document/cons_doc_law_34683. Дата обращения: 25.04.2017 г.
54. Русак О.Н., Малаян К.Р., Занько Н.Г., Безопасность жизнедеятельности: учебное пособие / Лань: 2000;
55. ГОСТ Р 55090-2012 Ресурсосбережение. Обращение с отходами. Рекомендации по утилизации отходов бумаги.

Приложение А

Раздел на иностранном языке

Раздел 1 An analytical review

Студент:

Группа	ФИО	Подпись	Дата
8ИМ5А	Сыздыкова Гаухар Жумагулкызы		

Консультант проф. кафедры:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. ИСТ	Мирошниченко Е.А.	К.Т.Н.		

Консультант – лингвист кафедры ИЯИК:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Горбатова Т.Н.			

1 ANALYTICAL REVIEW

The localization of the number plate on the image is extremely important. The results of all subsequent stages depend on how accurately the number plate on the image is determined. There is a number of methods for finding number plates: splitting an image into blocks; comparison with template (the region most closely resembling the template is segmented as the required number plate); applying of morphological operators; Hough transformation, with which you can find various shapes in the image (circle, ellipse, straight); Finding image edges using Sobel [2], Canny, Robinson algorithms [3], and others; the Viola-Jones method [4]; methods based on the analysis of histogram images [5] and the identification of the most contrasting areas. One of the most effective approaches for solving the problem of localizing objects in an image is the contour analysis method [6] and the Viola-Jones method [4]. With the help of these methods, in a variety of works very good results were obtained for the detection of various complex objects. Very good results for the detection of various complex objects were obtained in a variety of works with the help of these methods. They have high efficiency [3].

1.1 The method of contour analysis

Contour analysis is one of the important and very useful methods of describing, storing, recognizing, comparing and searching for graphic images or objects.

Edges are curves in the image, along which sharp change in brightness occurs [7]. The contour is the outline of an object. In contour analysis, the contour describes and contains the necessary information about the shape of the object. In this approach, the interior points of the object are not attention. This reduces the time for data processing.

When selecting image outlines, a number of problems can arise:

- presence of contour ruptures in image areas, where the brightness does not change too quickly;

- the presence of false contours due to noise in the image;
- the presence of wide contour lines due to fuzziness or noise.

With the help of contour analysis allows to describe, store, compare and appear objects represented in the form of their outlines - contours, and also effectively solve the most basic tasks such as moving, rotating and zooming the image of an object.

In contour analysis, the contour is encoded by a sequence consisting of complex numbers. The initial point is fixed on the contour. Then the contour is bypassed (for example, in the direction of the clockwise movement), and each displacement vector is noted by a complex number $a + ib$, where a – is the point displacement on the x axis; b – is the point displacement on the y axis. Displacement is noted relative to the previous point (figure 1.1) [8].

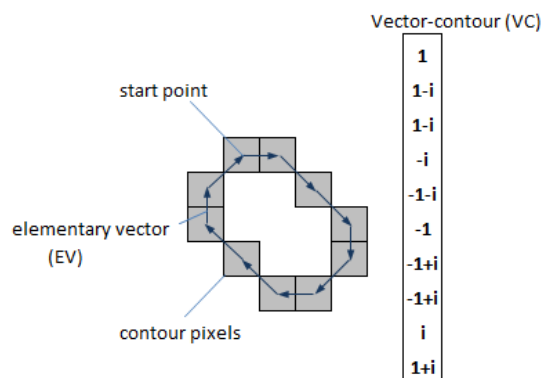


Figure 1.1 – Contour encoding

When objects are detected by the method of contour analysis the sequence of an operation is as follows:

1. Pre-processing of the image.
2. Selection of the edges of objects.
3. Finding image contour.
4. Work with contour parameters.

1.1.1 Canny edge detector

The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. It was developed by John

F. Canny in 1986. Canny also produced a computational theory of edge detection explaining why the technique works.

The main stages of the algorithm are:

3. Smoothing. Blur the image in order to remove noise. The Canny operator uses a filter that can be well approximated to the first Gaussian derivative. $\delta = 1.4$:

$$B = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} * A. \quad (1.1)$$

4. The search for gradients occurs in this way: boundaries are fixed in places, where the gradient of the image reaches its maximum value. The directions of the gradients can be different. Therefore, in the Canny algorithm four filters are used to detect horizontal, vertical and diagonal edges in a blurry image. Determining the value of the first derivative in the horizontal (G_y) and vertical directions (G_x) occurs with the use of edge detection operators. An example, by using the Sobel operator [9].

$$G = \sqrt{G_x^2 + G_y^2}. \quad (1.2)$$

The angle of the edge direction is calculated using the gradient:

$$\theta = \arctg\left(\frac{G_y}{G_x}\right). \quad (1.3)$$

The angle of the edge direction is rounded to one of the four corners representing the vertical, horizontal and two diagonals.

3. When suppressing non-maxima, the gradient is checked for the fact that in the corresponding direction it reaches a local maximum.

Points that have reached a local maximum are selected for grouping in the edge as the candidates.

An example for a 3x3 grid is as follows:

- if the gradient angle is zero, the point will be considered an edge if its intensity is greater than the point above and below the point under consideration,

- if the gradient angle is 90 degrees, the point will be considered an edge if its intensity is greater than the point on the left and the right of the point under consideration,

- if the gradient angle is equal to 135 degrees, the point will be considered an edge if its intensity is greater than that of the points in the upper left and lower right of the point in question,

- if the gradient angle is 45 degrees, the point will be considered an edge if its intensity is greater than that of the points in the upper right and lower left corner of the point under consideration [9].

The result is an image containing edges.

4. Double threshold: potential edges are determined by establishing a threshold.

5. Final edges: edges unconnected with strong boundaries are suppressed and the final boundaries are set up.

1.1.2 The Douglas-Peucker algorithm for approximating contours

The Douglas- Peucker algorithm is an algorithm that allows you to reduce the number of points of a curve approximated by a larger series of points. The algorithm was independently discovered by Urs Ramer in 1972 and by David Douglas and Thomas Peucker in 1973 [10].

The input of the Douglas-Peucker algorithm is an approximated contour and a given distance $\epsilon > 0$. ϵ – this is the distance between the first and last point. It was found that to achieve the best result of approximation ϵ , it makes sense to take $0.02 \times L$, where L - is the length of the approximated contour.

The algorithm divides the line recursively. A step-by-step explanation of the algorithm is as follows:

Start with a straight line going from the start point to the end point. If the contour is closed, choose points as far apart as possible. The step-by-step results shown in figures 1.2 - 1.5.

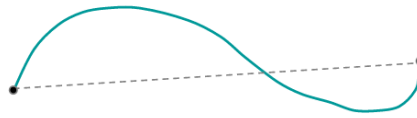


Figure 1.2 – The progress of the Douglas-Pequer algorithm

Step 1 - find the contour point as far as possible from the straight line. If the distance from it to the spaced $d < \epsilon$ - the partition is complete, if not - to step 2.

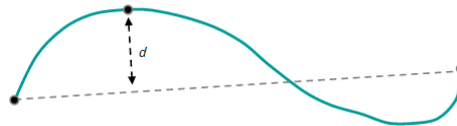


Figure 1.3 – The first iteration of the Douglas-Picker algorithm's work

Step 2 - add a node to the polyline. Then recursively call Step 1 for each of the halves of the polyline.

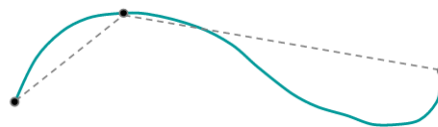


Figure 1.4 –The second iteration of the Douglas-Picker algorithm's work

The obtained result is as follows:

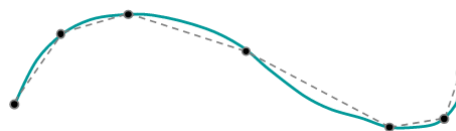


Figure 1.5 – The third iteration of the Douglas-Picker algorithm's work

The output polyline is constructed upon completion of all recursive calls, from points marked to the preservation [10].

1.2 The method of Viola-Jones

The method of Viola-Jones – an algorithm that allows to detect objects on images in real time, was proposed in 2001 by P. Viola and M. Jones [11].

Currently, the method of Viola-Jones is a famous method for detecting an object in an image due to its high speed and efficiency. The basic principles of the Viola-Jones method are: integral representation of the image, Haar features, construction of a classifier based on the boosting algorithm and a method of combining classifiers into a cascade structure. These principles make it possible to search for an object in real time.

1.2.1 Integral image

Integral representation of images is used to perform certain actions with data. The integral representation of the image is a matrix that is the same size with the primary image. The total brightness of all pixels located to the left and above the current element - the right bottom corner of the rectangular area (0,0) to (x, y) is stored in each element of the matrix. By the following formula, we can calculate the elements of the matrix L [4]:

$$L(x, y) = \sum_{i=0, j=0}^{i \leq x, j \leq y} I(i, j), \quad (1.4)$$

where $I(i, j)$ – pixel brightness of the original image.

Calculation of the values of matrix elements passes in a time proportional to the number of pixels in the current image. Thus, the integral image is read in one pass. The matrix elements are counted according to the formula:

$$L(x, y) = I(x, y) - L(x - 1, y - 1) + L(x, y - 1) + L(x - 1, y), \quad (1.5)$$

The sum of the pixels of an arbitrary rectangular area of arbitrary area can be calculated very quickly after the integration matrix is exposed.

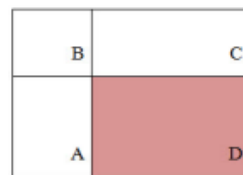


Figure 1.6 – Rectangle with areas A , B , C and D

In the rectangle $ABCD$ in figure 1.6 there is an object of interest to us D . The sum of the pixels in the rectangle D can be expressed by the sum and difference of the neighboring rectangles by the following formula:

$$S(D) = I(B) + I(ABCD) - I(BA) - I(BC). \quad (1.6)$$

With the integrated image representation, you can quickly calculate the total brightness of an arbitrary rectangular area in the image.

1.2.2 Haar features

In the Viola-Jones method is a window of a certain size, which moving along the image, is used at the stage of object detection. The Haar feature, through which the desired object is searched, is calculated in the areas of the image where the window passes [5].

Feature-display $f: X \Rightarrow D_f$ where D_f is the set of admissible values of the feature. The vector of feature $x = (f_1(x), \dots, f_n(x))_{x=(f_1(x), \dots, f_n(x))}$ is called a feature description of the object x , if the features are set f_1, \dots, f_n . Feature descriptions can be associated with the objects themselves. Moreover, the set $X = D_{f_1} * \dots * D_{f_n}$ is called the feature space.

The characteristics are divided into the following types, depending on the set D_f [12]:

- binary: $D_f = \{0, 1\}$;
- nominal: D_f — is a finite set;
- ordinal: D_f — is a finite ordered set;
- quantitative: D_f — set of real numbers.

The Haar features are calculated from adjacent rectangular areas. The Viola-Jones are used rectangular primitives. Primitives are shown in the figure 1.7.

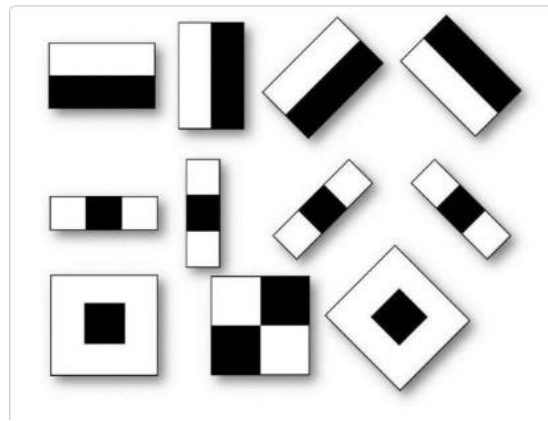


Figure 1.7 – Haar features

The value of the Haar feature F is calculated by the following formula:

$$F = X - Y, \quad (1.7)$$

where X – the sum of the brightness values of the points covered by the light part of the feature, Y – the sum of the brightness values of points covered by the dark part of the feature [13]. The calculations are performed using the integral image discussed above, and Haar primitives can be counted very quickly, for a constant time.

The Haar features are not very suitable for training and classification. More features are required to characterize an object with sufficient correctness. Therefore, the Haar signs come to the cascade classifier, which serves for quick discarding of the windows where the required object was not found, and producing the result "true" or "false" according to the location of the object.

1.2.3 Classifier based on adaptive boosting algorithm

The classifier in classification problems is an approximating function that decides which category the object belongs to. The classifier is built on the basis of the boosting algorithm in order to select the most appropriate features for the searched object in a certain area of the image.

Boosting – this is a set of methods that contribute to increasing the accuracy of classification. Models in which a classification error is allowed in a smaller amount are called "strong". Models in which a classification error is allowed in in

large numbers are called "weak". A weak model does not allow you to reliably separate classes or give accurate predictions, makes a large number of errors. In this connection, boosting means "strengthening" of "weak" models. Boosting is a procedure for the sequential construction of a composition of machine learning algorithms, where each subsequent algorithm seeks to compensate for the weakening of the composition of all previous algorithms.

A simple classifier is formed at each iteration at the output of the boosting algorithm:

$$h_{j(z)} = \begin{cases} 1, & \text{если } p_i f_j(z) < p_i \theta_j \\ 0, & \text{иначе} \end{cases} \quad (1.8)$$

where p_i — direction of the inequality sign, θ_j — threshold value, $f_j(z)$ – counted value of the feature, z – image window size 24 * 24 pixels [13].

The output classifier has a minimum error regarding the current values of the weights that are involved in the learning process for finding errors.

A trained classifier, which represented in the xml format, is used to search for an object on a digital image. The classifier is formed on the Haar [15]. An example of a classifier structure is shown in figure 1.8.

```

<maxWeakCount>6</maxWeakCount>
<stageThreshold>-1.3110191822052002e+000</stageThreshold>
<weakClassifiers>
  <_>
    <internalNodes>
      0 -1 193 1.0079263709485531e-002</internalNodes>
    <leafValues>
      -8.1339186429977417e-001 5.0277775526046753e-001</leafValues></_>
  <_>
    <internalNodes>
      0 -1 94 -2.2060684859752655e-002</internalNodes>
    <leafValues>
      7.9418992996215820e-001 -5.0896102190017700e-001</leafValues></_>

```

Figure 1.8 – Classifier structure

where $\langle \text{maxWeakCount} \rangle$ – total number of weak classifiers; $\langle \text{stageThereshold} \rangle$ – maximum brightness threshold; $\langle \text{weakClassifiers} \rangle$ – a set of weak classifiers. They decide whether there is an object on the image or not; $\langle \text{internalNodes} \rangle$ и $\langle \text{leafValues} \rangle$ – the characteristics of a specific weak classifier.

A cascade of classifiers that decides whether an object is recognized on the image or not is built on this basis. The presence or absence of an object in the window is determined by the difference between the value of feature and the threshold, which received as a result of training [15].

1.2.4 Training scheme

For the algorithm it is necessary to prepare in advance a test sample containing the required object and m - not containing. Then the number of all test images will be [16]:

$$n = l + m, \quad (1.9)$$

$$X = \{ x_1, x_2, \dots, x_n \}, \quad (1.10)$$

where X – set of all test images, where for each in advance it is known whether the required object is present or not and is reflected in the set Y .

$$Y = \{ y_1, y_2, \dots, y_n \}, \quad (1.11)$$

where

$$y_1 = \begin{cases} 1, \text{object is present in the image } x_1 \\ 0, \text{else} \end{cases}. \quad (1.12)$$

The feature j will be understood as the structure of the following type:

$$j = \{ \text{маска, положение, размер} \}. \quad (1.13)$$

Then the feature response is $f_j(x)$, which is calculated as the difference in pixel intensities in the light and dark areas. A weak classifier is represented by the following formula:

$$h_j(x) = \begin{cases} 1, p_j f_j(x) < p_j \theta_j \\ 0, \text{else;} \end{cases} \quad (1.14)$$

The task of a weak classifier is to guess the presence of an object in more than 50% of cases. A strong classifier is created using the AdaBoost training procedure, which consisting of T weak classifiers and represented by the following formula:

$$H(x) = \begin{cases} 1, & \sum_{t=1}^T a_t h_{j^{(t)}}(x) \geq \frac{1}{2} \sum_{t=1}^T a_t \\ 0, & \text{иначе} \end{cases} \quad (1.15)$$

The target learning function has the following form:

$$T, h_{j^{(1)}}, h_{j^{(2)}}, \dots, a_1, \dots, a_T = \underset{i=1}{\operatorname{argmin}} \sum_{i=1}^n |H(x_i, T, h_{j^{(1)}}, \dots, h_{j^{(T)}}, a_1, \dots, a_T)|. \quad (1.16)$$

Weights $w_{q,i}$ are initialized before the training begins, where q – iteration number, i - image number.

$$w_{1,i} = \begin{cases} \frac{1}{2l}, & y_i = 1 \\ \frac{1}{2m}, & y_i = 0 \end{cases}. \quad (1.17)$$

T weak classifiers and T values will be obtained after the training procedure.

1.3 Methods of the image smoothing

Preprocessing is the transformation of the original image into some other, improved image. Of course, the most popular way of pre-processing images is to filter [17], which is used to remove noise in most applications. A new image of the same size as the original image is obtained as a result of filtering. The new image contains pixel intensity values that are updated according to some rule. There are many methods of smoothing images, differing in their parameters and output results. These filters are linear, Gaussian filter, nonlinear, median, bilateral, ranging, adaptive, combined, hybrid etc.

Digital images are exposed to various types of noise. They can arise in the process of obtaining images, transferring information and digitizing data. Filtering is the process of eliminating various types of noise in the image.

During the filtering process, the brightness parameters of each point of the

digital image are replaced by the brightness values which is recognised as the least distorted interference [18, 19].

There are frequency and spatial methods of filtration. Frequency methods of image transformations [19-24] are based on the Fourier transform idea. In the Fourier transform, the original function is represented as a sum of trigonometric functions of different frequencies, which are multiplied by specific coefficients. The function represented by the Fourier transform can be reverted back to the original form after the transformations. The implementation of frequency approaches can be similar to spatial methods of filtration [18].

Spatial methods for improving images are applied to raster images. Special operators are applied to each point of the image, in such spatial algorithms. The operators are rectangular or square matrices. They are called masks, cores or windows [18, 25, 26]. In many cases, the mask appears to be a small two-dimensional array, and improvement methods based on this aspect often called mask processing or mask filtering.

1.3.1 Linear filters

Linear filters are the simplest representatives of spatial methods of preprocessing. Linear filtering reduces to recalculating the intensity values of each pixel of the image by calculating the weighted convolution of the pixel intensities belonging to some of its neighborhoods. Blurring or smoothing is another approach to preprocessing. Smoothing, like linear filtering, in the simplest case involves convolution with equal weight coefficients. In more complex applications it involves convolving with discrete values of the Gaussian distribution function or choosing a median among a set of intensities in the neighborhood [27].

The response of the mask is specified by the sum of the pixel derivatives in the filter coverage area in the linear filtering. The averaging filter is used as a linear smoothing filter. The output value of the averaging filter is the average value along the round of the filter mask [18]. A filter of this kind is used to remove the graininess

of the image, which is caused by impulse noise. The response $g(x, y)$ of the averaging filter, which calculated to filter the image f with the sizes $M \cdot N$ represented by the following formula:

$$g(x, y) = \sum_{s=\frac{(m-1)}{2}}^{(m-1)/2} \sum_{t=\frac{(n-1)}{2}}^{(n-1)/2} w(s, t) f(x + s, y + t), \quad (1.18)$$

where $w(s, t)$ – element of the convolution kernel, with size $m \times n$; $s \in \left[-\frac{m}{2}, \frac{m}{2}\right]$ – kernel coordinates along the abscissa, $t \in \left[-\frac{n}{2}, \frac{n}{2}\right]$ – coordinates along the ordinate axis; $x = 0, 1, 2, \dots, M - 1$, $y = 0, 1, 2, \dots, N - 1$ – image coordinates f .

Such a filter, for a convenient software representation, is expressed in the following formula:

$$G_{i,j} = \sum_{s=\frac{(m-1)}{2}}^{(m-1)/2} \sum_{t=\frac{(n-1)}{2}}^{(n-1)/2} w_{s,t} \cdot E_{(i+s,j+t)}, \quad (1.19)$$

where $G_{i,j}$ – element of the image matrix after applying the filtering; $[w_{s,t}]$ – array element of the image convolution kernel, with size $m \times n$; $[E_{i,j}]$ – element of the image matrix.

1.3.2 Gaussian filter

The use of the Gaussian filter allows smooth and reduce the noisiness of the image. The kernel of this filter can be expressed by the following formula:

$$F_{gauss}(i, j) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{i^2 + j^2}{2\sigma^2}\right), \quad (1.20)$$

where i, j – image pixel coordinates; f – signal, a σ – noise on the original image.

The Gaussian filter blurs the noise by exposing the content contours of the image to a small degree of blur. For example, if you need to blur small details on the source image, which do not require separation from the background, and we will allocate large objects of interest in the future using binarization.

Nonlinear filters

Nonlinear spatial methods are analogous to linear filters according to the principle of operation. The values of the matrix elements of the original image play an important role. They are analyzed by the circle and the operations performed by the nonlinear filter depend on them. An example of the work of the nonlinear filter is to calculate the median of the values of the elements of the circle being analyzed [18].

Median Filter

The pixel values represent the averaged value of the points of the corresponding neighborhood in the median filtering. The median filter is more efficient than conventional averaging when solving noise elimination problems, in view of the fact that it leads to less distortion of the boundaries of objects on the isobasis. The median filter uses a two-dimensional window with central symmetry as a mask and its center is located at the current filtration point [19]. Two examples of the most commonly used variants of windows: a) in the form of a cross; B) in the form of a square are shown in the figure 1.9. The size of windows is optimized in the process of analyzing the effectiveness of the algorithm. The counts of the original image that appeared within the window, contribute to the formation of a working sample of the current step.



Figure 1.9 – Examples of windows with median filtering

The two-dimensional median filter is defined by the following formula:

$$G_{i,j} = \text{med}[E_{(i+s,j+t)}; (s,t) \in w]; i,j \in Z^2, \quad (1.21)$$

where $G_{i,j}$ – element of the image matrix after filtering; $[w_{s,t}]$ – array element of im-

age windows, with size $m \times n$; $[E_{i,j}]$ – matrix element of the original image.

The bilateral filter

The bilateral filter smoothes the image by keeping the borders clear [28]. The bilateral filter calculates the intensity of each pixel as a weighted mean of the neighboring pixel intensities. This is the result of his work [18]. The weight associated with each adjacent pixel changes in accordance with the values of the distances in the image plane (spatial weight) and on the intensity axis (rank weight). This allows you to keep clear boundaries of image elements.

The results depend only on the distance between the pixels. This is the main idea of this method.

The image is represented in three-dimensional space, where the coordinate of the pixel intensity is added to the standard coordinates of the image plane, that is, we calculate the homogeneous vector by the formula:

$$\begin{pmatrix} I_p \cdot W_p \\ W_p \end{pmatrix} = \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(|I_p - I_q|) \begin{pmatrix} I_q \\ 1 \end{pmatrix}, \quad (1.22)$$

where: S – the set of possible pixel coordinates of the image; R – number of colors involved; $p = (p_x, p_y) \in S$ – the position of the central pixel; q – adjacent pixel coordinate, $I_p, I_q \in R$ – the intensity of the pixels p and q in the range $[0, 1]$; $G_{\sigma_s}, G_{\sigma_r}$ – Gauss function with parameters σ_s and σ_r ; W_p – average of weights for normalization of values [29].

The number of neighboring pixels is specified by the parameter σ_s , the weight loss by neighboring pixels is given by σ_r .

The generated space is compressed with the sampling rate set, while keeping the information for each pixel in its corresponding storage cell. The large structure consisting of such cells is called the bilateral lattice. Further, the processing with the bilateral filter is performed taking into account only the gap between the pixels, in particular, coagulation with a three-dimensional Gaussian function with properties

$\frac{\sigma_s}{S_s}, \sigma_r$
 $\frac{S_s}{S_r}$. The result is restored to the original pixel positions: three-line interpolation and normalization of the result. The original image shown in figure 1.10.



Figure 1.10 – The original image

The example of image filtering using a homogeneous (a), Gaussian (b), median (c) and bilateral filter (d) shown in figure 1.11.

a) Homogeneous filtration



b) Gaussian filtering



c) Median filtration



d) Bilateral filtering



Figure 1.11 – The result of the image filtering

Приложение Б

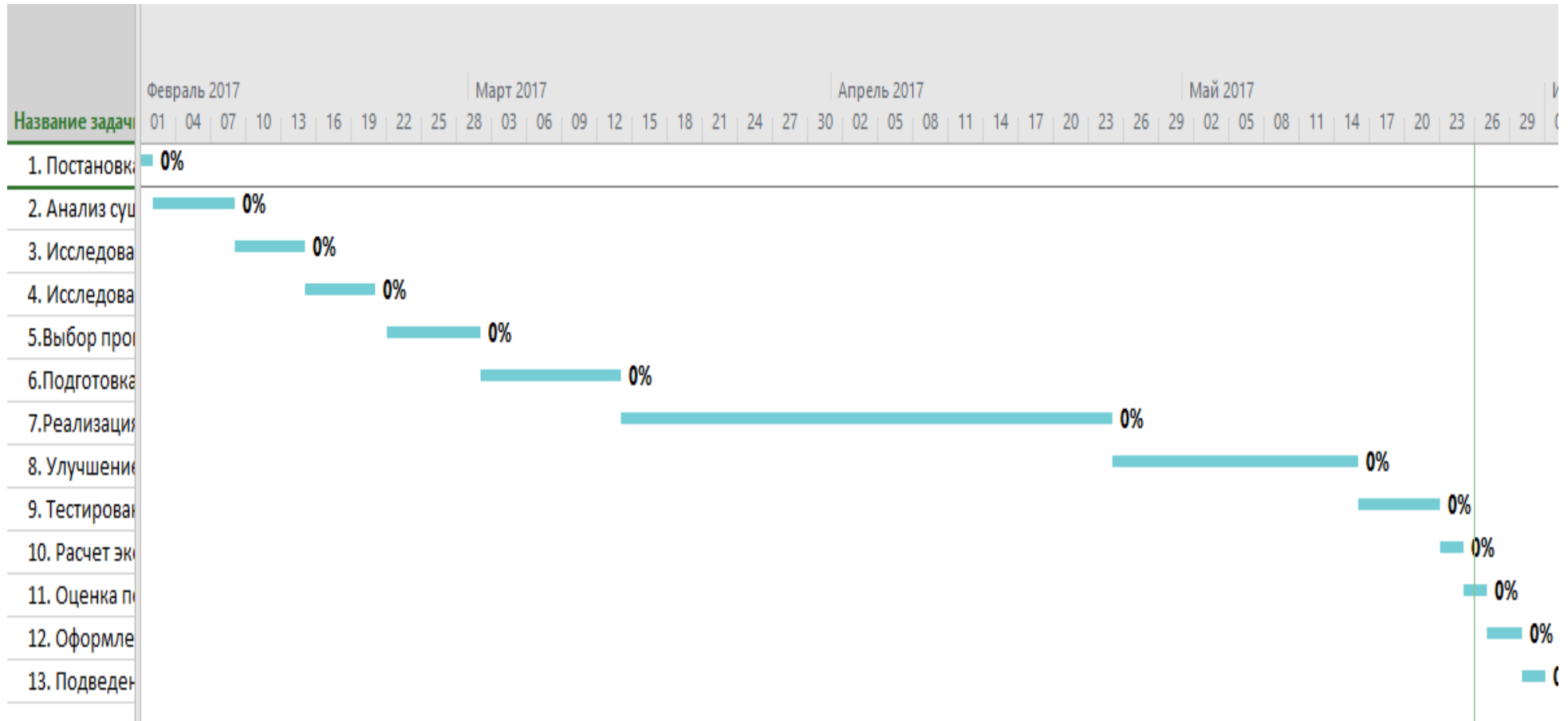


Рисунок 1 – Диаграмма Ганта для исполнителя проекта

