

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

Инженерная школа ядерных технологий
Направление подготовки: 01.04.02 «Прикладная математика и информатика»
Отделение экспериментальной физики

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Разработка алгоритма и программы распознавания автомобильного регистрационного номерного знака на локализованной области

УДК 004.032.26

Студент

Группа	ФИО	Подпись	Дата
0ВМ82	Кенджаев Голибджон Орифович		

Руководитель ВКР

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

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Рыжакина Татьяна Гавриловна	к.э.н., доцент		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Романова Светлана Владимировна			

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Семенов Михаил Евгеньевич	к.ф.-м.н., доцент		

Планируемые результаты освоения по ООП

Код результата	Результаты обучения
	Общепрофессиональные компетенции
P1	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.
	Демонстрировать способность к самостоятельному обучению новым методам исследования, способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения.
P5	Использовать на практике умения и навыки в организации исследовательских работ по использованию математических методов и моделирования для решения задач, способность организовывать и руководить работой команды.
P6	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.
	Профессиональные компетенции
P7	Разрабатывать стратегии и цели проектирования, критерии эффективности и ограничения применимости, новые методы, средства и технологии использования математических методов в экономике, науке и технике.
P8	Планировать и проводить теоретические исследования и компьютерные эксперименты при использовании математических методов в экономике, науке и технике.

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

Инженерная школа ядерных технологий

Направление подготовки 01.04.02 «Прикладная математика и информатика»

Отделение экспериментальной физики

УТВЕРЖДАЮ:

Руководитель ООП

_____ Семенов М.Е.

(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

В форме:

Магистерской диссертации

Студенту:

Группа	ФИО
0ВМ82	Кенджаеву Голибджону Орифовичу

Тема работы:

Утверждена приказом директора (дата, номер)	№133-38/с от 12.05.2020

Срок сдачи студентом выполненной работы:	31.05.2020
--	------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p>Исходные данные к работе <i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.);</i></p>	<p>Объект исследования – алгоритм для выделения границ и распознавания автомобильных регистрационных номерных знаков на графическом изображении. Программное обеспечение осуществляет сегментацию и распознавание символов номерных знаков автомобилей.</p>
<p>Перечень подлежащих исследованию, проектированию и разработке вопросов <i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<ol style="list-style-type: none"> 1. Обзор и анализ основных методов сегментации и распознавания автомобильных номерных знаков. 2. Разработка алгоритма распознавания автомобильных номерных знаков. 3. Программная реализация алгоритма распознавания. 4. Разработать основные компоненты программного обеспечения и графический интерфейс пользователя для работы с программной реализацией алгоритма распознавания номерных знаков. 5. Сделать выводы по проделанной работе.
<p>Перечень графического материала <i>(с точным указанием обязательных чертежей)</i></p>	<ol style="list-style-type: none"> 1. Блок-схема алгоритма. 2. Структура основных компонентов программного обеспечения. 3. Изображения работы алгоритма сегментации и распознавания номерных знаков, интерфейс программного обеспечения.
<p>Консультанты по разделам выпускной квалификационной работы <i>(если необходимо, с указанием разделов)</i></p>	
<p>Раздел</p>	<p>Консультант</p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Рыжакина Татьяна Гавриловна rigakina@tpu.ru</p>
<p>Социальная ответственность</p>	<p>Романова Светлана Владимировна romanova@tpu.ru</p>

<p>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</p>	
--	--

Задание выдал руководитель:

<p>Должность</p>	<p>ФИО</p>	<p>Ученая степень, звание</p>	<p>Подпись</p>	<p>Дата</p>
<p>Доцент</p>	<p>Семенов М.Е.</p>	<p>к.ф.-м.н., доцент</p>		

Задание принял к исполнению студент:

<p>Группа</p>	<p>ФИО</p>	<p>Подпись</p>	<p>Дата</p>
<p>0ВМ82</p>	<p>Кенджаев Голибджон Орифович</p>		

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

Студенту:

Группа	ФИО
0BM82	Кенджаев Голибджон Орифович

Школа	ИЯТШ	Отделение школы (НОЦ)	ОЭФ
Уровень образования	Магистр	Направление/специальность	01.04.02 Прикладная математика и информатика

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

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Использование информации, представленной в российских и иностранных научных публикациях, аналитических материалах и изданиях, нормативно-правовых документах
2. <i>Нормы и нормативы расходования ресурсов</i>	
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого и инновационного потенциала НТИ</i>	1. <i>Потенциальные потребители результатов исследования;</i> 2. <i>Анализ конкурентных технических решений;</i>
2. <i>Разработка устава научно-технического проекта</i>	1. <i>Структура работ в рамках научного исследования;</i>
3. <i>Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок</i>	1. <i>Определение трудоемкости выполнения работ и разработка графика проведения научного исследования;</i> 2. <i>Бюджет научно - технического исследования (НТИ).</i>
4. <i>Определение ресурсной, финансовой, экономической эффективности</i>	<i>Проведение оценки ресурсной, финансовой, экономической эффективности научного исследования</i>

Перечень графического материала(с точным указанием обязательных чертежей):

1. *График проведения и бюджет НИР*
2. *Оценка ресурсной, финансовой и экономической эффективности НИР*

Дата выдачи задания для раздела по линейному графику 31.01.2020

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Рыжакина Татьяна Гавриловна	к. э. н.		31.01.2020

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0BM82	Кенджаев Голибджон Орифович		31.01.2020

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
0BM82	Кенджаев Голибджон Орифович

Школа	ИЯТШ	Отделение (НОЦ)	ОЭФ
Уровень образования	Магистратура	Направление/ специальность	01.04.02 Прикладная математика и информатика

Тема ВКР:

Разработка алгоритма и программы распознавания автомобильного регистрационного номерного знака на локализованной области	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объект исследования - численные методы решения обыкновенных дифференциальных уравнений второго порядка. Область применения: применяются для математического описания природных явлений.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности:	<ul style="list-style-type: none"> – «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» (СанПиН 2.2.2/2.4.1340-03); – ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования»; – «Трудовой кодекс Российской Федерации» от 30.12.2001 N 197-ФЗ (ред. от 16.12.2018)
2. Производственная безопасность:	<ul style="list-style-type: none"> – Отклонение показателей микроклимата; – Превышение уровня шума; – Недостаточная освещенность рабочей зоны; – Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека; – Повышенный уровень

	электромагнитных излучений.
3. Экологическая безопасность:	Влияние объекта исследования на окружающую среду. – Утилизация компьютерной техники; – Утилизация бумаги, использованной в процессе разработки.
4. Безопасность в чрезвычайных ситуациях:	Чрезвычайная ситуация техногенного характера для данного помещения – пожар. Установка общих правил поведения и рекомендаций во время пожара, план эвакуации.

Дата выдачи задания для раздела по линейному графику	23.03.2020г.
---	--------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Романова С.В.			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0BM82	Кенджаев Г.О.		

РЕФЕРАТ

Выпускная квалификационная работа содержит: 124 листов, 23 рисунков, 32 таблицы, 44 источника литературы.

Ключевые слова: сверточная нейронная сеть, автомобильный номер, распознавание номерных знаков, программное обеспечение, алгоритмы, сегментация.

Объект исследования: системы компьютерного зрения, технология распознавания автомобильных номеров, которая основывается на алгоритмах обработки изображений и распознавания символов.

Цель работы: разработка и программная реализация алгоритма распознавания номерных знаков транспортных средств на полученном локализованном изображении.

Работа включает: введение, пять разделов, список публикаций студента, а также список использованных литературных источников.

В разделе 1 «Литературный обзор предметной области» приведен краткий обзор литературы, необходимый для дальнейшей работы, сформулирована цель и задачи работы.

В разделе 2 «Программная реализация алгоритма» приведено описание процесса разработки, реализован алгоритм и программа распознавания автомобильного регистрационного номерного знака на локализованной области.

В разделе 3 «Результаты тестирования программного обеспечения» приведены результаты тестирования программного обеспечения и таблица сравнения разработанного алгоритма с существующими аналогами.

В разделе 4 «Социальная ответственность» оценены риски и опасности для здоровья и окружающей среды и рассмотрены способы их минимизации.

В разделе 5 «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» рассмотрена экономическая составляющая работы. Посчитаны затраты на реализацию и изучены конкуренты на рынке, а также посчитана рентабельность проекта.

ОПРЕДЕЛЕНИЯ И ИСПОЛЬЗОВАННЫЕ СОКРАЩЕНИЯ

ГРЗ – государственный регистрационный знак

КПП – контрольно-пропускной пункт.

Application programming interface (API) – интерфейс программирования приложений, позволяющий сервисам взаимодействовать, получать доступ и обмениваться данными.

Automatic number-plate recognition (ANPR) — автоматическое распознавание номерных знаков.

Convolutional Neural Network (CNN) – сверточная нейронная сеть.

OpenCV (Open Source Computer Vision Library) – библиотека компьютерного зрения

Optical Character Recognition (OCR) – оптическое распознавание символов.

Rivest–Shamir–Adleman (RSA) – один из первых криптосистем с открытым ключом и широко используется для безопасной передачи данных.

XML (eXtensible Markup Language) – язык гипертекстовой разметки.

Оглавление

ВВЕДЕНИЕ.....	13
1 ЛИТЕРАТУРНЫЙ ОБЗОР ПРЕДМЕТНОЙ ОБЛАСТИ	14
1.1 Введение в предметную область.....	14
1.2 Характеристики систем распознавания автомобильных номеров	16
1.3 Алгоритмы и методы обработки изображений	17
1.4 Методы сегментации	20
1.5 Библиотеки для обработки изображений	25
1.6 Выводы по литературному обзору	26
2 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМА.....	27
2.1 Архитектура программного обеспечения.....	27
2.2 Выделение номерной рамки.....	28
2.3 Решение задачи распознавания автомобильного номера	31
2.4 Графический интерфейс пользователя.....	34
3 РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.....	36
3.1 Тестирование программного продукта.....	36
3.2 Тестирование алгоритма распознавания	36
4 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ	38
4.1 Правовые и организационные вопросы обеспечения безопасности	38
4.1.1 Правовые нормы трудового законодательства для рабочей зоны.....	38
4.1.2 Производственная безопасность.....	39
4.2 Анализ вредных и опасных факторов, которые может создать объект исследования. 40	
4.2.1 Отклонение показателей микроклимата.....	40
4.2.2 Повышенный уровень шума.....	42
4.2.3 Недостаточная освещенность рабочей зоны.....	43
4.2.4 Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека	45
4.2.5 Повышенный уровень электромагнитных излучений	47
4.3 Экологическая безопасность.....	48
4.4 Безопасность при чрезвычайных ситуациях	49
5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ 53	
5.1 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....	53
5.2 Предпроектный анализ.....	53
5.2.1 Потенциальные потребители результатов исследования	53

5.2.2	Анализ конкурентных технических решений	54
5.2.3	<i>SWOT</i> -анализ.....	55
5.2.4	Диаграмма Исикавы.....	58
5.2.5	Определение возможных альтернатив проведения исследования.....	59
5.3	Оценка готовности проекта к коммерциализации.....	60
5.3.1	Методы коммерциализации результатов научно-технического исследования..	61
5.4	Инициализация проекта	62
5.4.1	Цели и результат проекта.....	63
5.4.2	Организационная структура проекта	64
5.5	План работ по научно-исследовательскому проекту	65
5.5.1	Структура работ в рамках научного исследования	65
4.6	Бюджет научного исследования	68
5.5.2	Расчет затрат на специальное оборудование для научных (экспериментальных) работ	68
5.5.3	Основная заработная плата	69
5.5.4	Дополнительная заработная плата научно-производственного персонала	71
5.5.5	Отчисления на социальные нужды	71
5.5.6	Прочие прямые затраты	72
5.5.7	Накладные расходы.....	73
5.5.8	Формирование бюджета затрат НИИ	73
5.6	Реестр рисков проекта.....	74
5.7	Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.....	76
5.7.1	Оценка абсолютной эффективности исследования.....	76
5.8	Оценка сравнительной эффективности исследования.....	82
	ЗАКЛЮЧЕНИЕ	86
	Список публикаций студента.....	87
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	88
	Приложение А	92
	Приложение В	105

ВВЕДЕНИЕ

В настоящее время компьютеризация в нашем обществе развивается довольно быстрыми темпами и играет большую роль в жизни человека. При помощи компьютерных технологий автоматизируется широкий круг задач и процессов, которые в минувшем прошлом возлагались на человека.

Данная магистерская диссертация посвящена разработке и программной реализации алгоритма распознавания автомобильных номеров. Задачу распознавания автомобильных знаков можно разделить на два этапа: поиск номерного знака и распознавание символов номерного знака, которое в свою очередь включает: а) предобработку изображения, б) сегментацию и в) распознавание символов.

Предметом исследования являются методы искусственного интеллекта, используемые в алгоритмах распознавания автомобильных номерных знаков, а также исследование возможности применения искусственных нейронных сетей в качестве классификаторов цифробуквенных символов автомобильных номерных пластин.

Объектом исследования является технология распознавания автомобильных номерных знаков на основе алгоритмов обработки изображений и распознавания символов.

Целью работы является разработка алгоритма распознавания номерных знаков на полученном локализованном изображении автотранспортных средств и разработка программы, использующей данный алгоритм. Для достижения поставленной цели необходимо последовательно решить следующие задачи:

- ознакомиться с методами распознавания символов на изображениях;
- разработать алгоритм для распознавания номерных знаков автотранспортных средств на полученном локализованном изображении;
- разработать программное обеспечение, использующее данный алгоритм.

1 ЛИТЕРАТУРНЫЙ ОБЗОР ПРЕДМЕТНОЙ ОБЛАСТИ

Существует большое количество промышленного программного обеспечения для распознавания автомобильных номеров. В данном разделе приведем краткий аналитический обзор программных продуктов. Описание каждого продукта содержит информацию о названии, области применения и используемых технологиях распознавания.

1.1 Введение в предметную область

Действующий государственный стандарт Российской Федерации определяет 22 типа государственных регистрационных знаков (ГРЗ), устанавливаемых на транспортные средства [1]. Стандарт определяет основные размеры, а также технические требования к ГРЗ. Описанные в стандарте технические требования направлены на обеспечение безопасности дорожного движения, жизни, здоровья, сохранности имущества населения и охраны окружающей среды.

В зависимости от применяемых регистрационных знаков транспортные средства под разделяют на следующие группы:

1 – транспортные средства, принадлежащие юридическим лицам и гражданам Российской Федерации, юридическим лицам и гражданам иностранных государств, кроме отнесенных к группе 3, а также лицам без гражданства;

2 – транспортные средства воинских частей и соединений, находящихся под юрисдикцией Российской Федерации и образованных в соответствии с действующими законодательными актами;

3 – транспортные средства, принадлежащие дипломатическим представительствам, консульским учреждениям, международным (межгосударственным) организациям;

4 – транспортные средства, временно допущенные к участию в дорожном движении;

5 – транспортные средства, принадлежащие органам внутренних дел Российской Федерации.

В данной работе рассматривается распознавание ГРЗ первого типа с трехзначным и двухзначным кодом региона согласно действующему стандарту РФ [1]. Стандарт устанавливает следующие типы регистрационных знаков (рисунок 1.1):

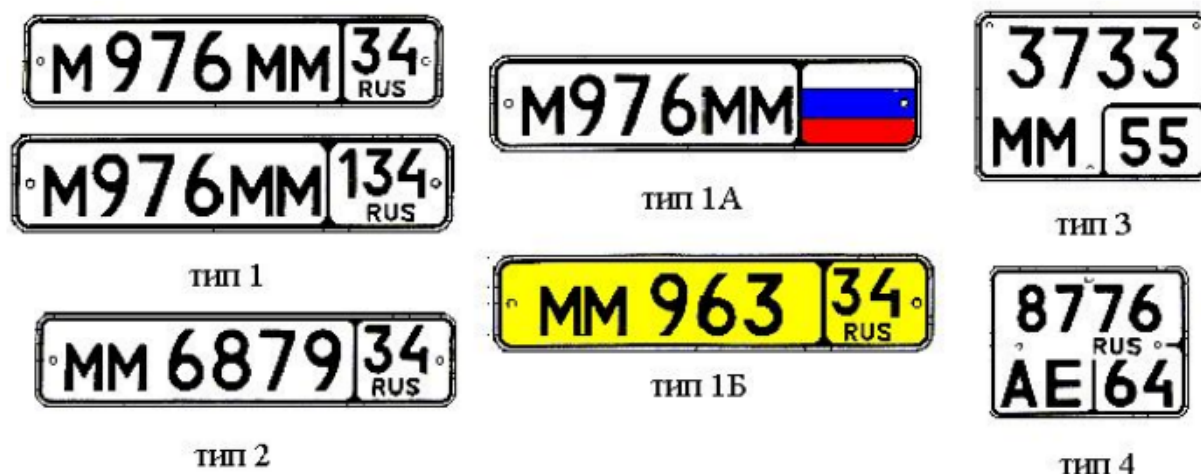


Рисунок 1.1 – Типы регистрационных знаков 1-й группы

Распознавание остальных типов ГРЗ в данной работе не рассматривается. Примеры ГРЗ типа 1 с двухзначным и трехзначным кодом региона регистрации показаны на рисунках 1.2 и 1.3.



Рисунок 1.2 – Знак типа 1 с двухзначным кодом региона [1]



Рисунок 1.3 – Знак типа 1 с трехзначным кодом региона [1]

Основная часть автомобильных номеров в России являются стандартными знаками образца 1993 года, определение которых содержит ГОСТ Р 50577-93 [1].

Автомобильные номерные знаки дипломатических транспортных средств, транспортных средств МВД, прицепов, военных транспортных средств имеют небольшое отличие от стандартного.

На стандартных номерных знаках РФ должны присутствовать: три буквы, три цифры, номер региона две или три цифры, флаг России и надпись RUS. Буквы означают серию номерного знака, а цифры - номер. Для использования на знаках разрешены 12 букв кириллицы, которые есть в латинском алфавите - А, В, Е, К, М, Н, О, Р, С, Т, У и Х.

Для каждого региона Российской Федерации имеется общий для всех транспортных средств, зарегистрированных в данном регионе, номер.

1.2 Характеристики систем распознавания автомобильных номеров

Среди программного обеспечения для распознавания автомобильных номеров можно выделить: «Автомаршал», «Спецлаб-Трафик», «SecurOSAuto». В таблице 1 приведем характеристики этих программ.

Таблица 1 - Системы распознавания автомобильных номеров

	Автомаршал	Спецлаб-Трафик	SecurOSAuto
Операционная система	Windows 7SP1/8.1/10 (32/64 бит) Windows Server 2008R2/2012/2012R2 /2016	Windows Server Linux Server	Windows Windows Server
Стоимость	29 900 руб/	30 000 руб.	48 110 руб.
Вероятность распознавания, %	98	99	98
Область применения	автоматизации работы парковок, КПП и проходных предприятий, учета транспорта на автомагистралях, весовых, автомойках и заправочных станциях, а также для управления доступом автомобилей на территорию многоквартирных жилых домов и загородных поселков	обеспечения безопасности, контроля въезда/выезда автотранспорта с автостоянок, коллективных гаражей, территорий с ограниченным доступом	перечень задач правоохранительных органов, включая розыск транспортных средств, автоматизация коммерческих и муниципальных парковок, контроль платных дорог, весовой контроль, контроль проезда транспорта на режимных объектах, пограничный и таможенный контроль

1.3 Алгоритмы и методы обработки изображений

Для обработки изображений существует множество методов и алгоритмов, приведем краткое описание использованных в работе методов и алгоритмов.

Фильтром Гаусса называется электронный фильтр, у которого функция Гаусса является импульсной переходной функцией. Фильтр Гаусса задуман так, чтобы постоянная времени была максимальной и отсутствовало

перерегулирование в функции перехода. Такое поведение тесно связано с тем, что фильтр Гаусса имеет минимально возможную групповую задержку. Фильтр Гаусса обычно используется в цифровом виде для обработки двумерных сигналов (изображений) с целью снижения уровня шума [20].

Следующий алгоритм — это **детектор Хаара**. Его основное преимущество — скорость. Быстрая обработка изображений позволяет легко обрабатывать видеопоследовательности. Детектор Хаара применяется для определения большинства классов объектов. К ним относятся лица и иные части тела людей, номера автомашин, прохожие, дорожные символы, животные и т.д.

Классификатор формируется на примитивах Хаара методом расчета характеристических значений. Для обучения на ввод классификатора подается комплект «правильных» изображений с предварительно избранной областью на изображении, вслед затем осуществляется поиск примитивов и вычисление значения признака [12]. В основном используются прямоугольные примитивы (рисунок 1.4).

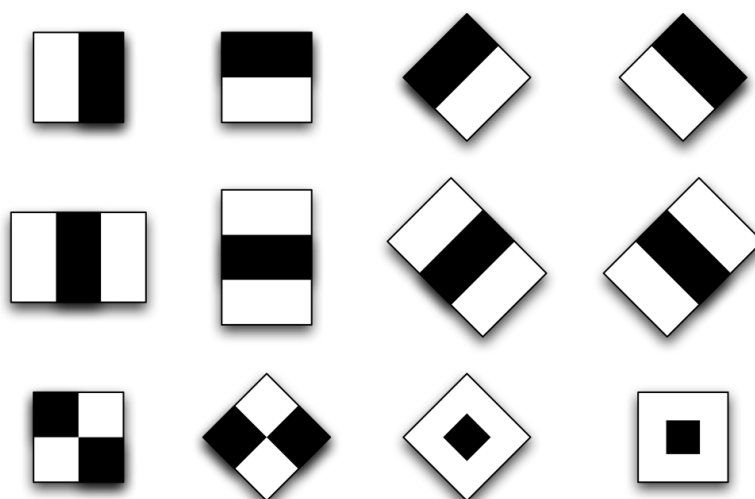


Рисунок 1.4 – Примитивы Хаара

В обзоре обратим внимание на метод гистограмм. построения **гистограмм яркости** изображения применяют операцию бинаризации – выделим чёрные и белые пиксели. Впоследствии бинаризации подсчитывается в каждой колонке численность чёрных пикселей и на базе

полученной информации строится гистограмма изображения. Принцип построения гистограммы отображён на рисунке 1.5.

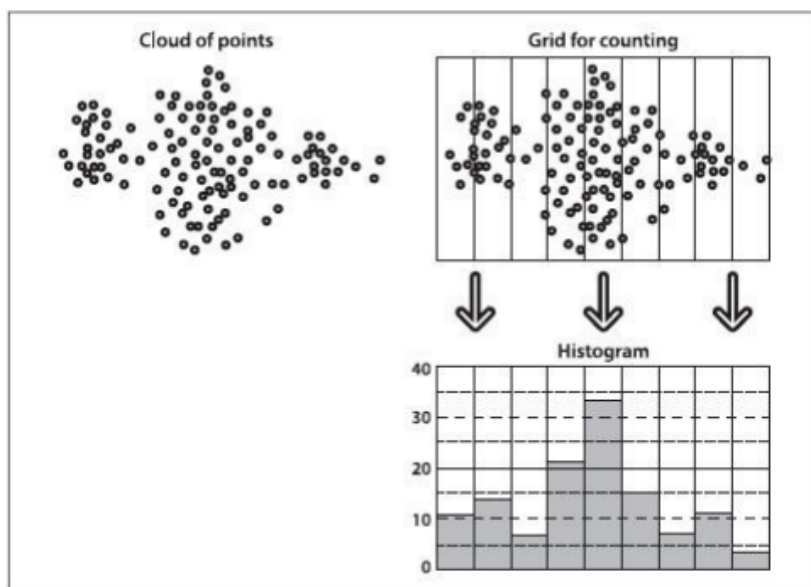


Рисунок 1.5 – Принцип построения диаграммы яркости

Пример бинаризованного автомобильного номера и его гистограмма (оранжевая линия) приведены на рисунках 1.6 и 1.7.



Рисунок 1.6 – Бинаризованный автомобильный номер

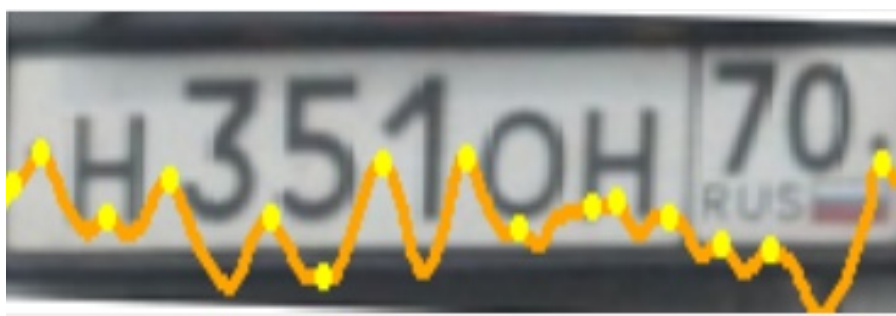


Рисунок 1.7 – Гистограмма яркости автомобильного номера

Из рисунка 1.7 видно, что имеются резкие колебания кривой, что используется для определения границ (сегментации) автомобильного номера.

1.4 Методы сегментации

Сегментация изображения — процесс разделения цифрового изображения на большое количество областей (различные объекты), которые имеют схожую текстуру или же цвет. Итогом сегментации считается набор областей (контуров), извлеченных из изображения. Все пиксели из одной области схожи по некоторым характеристикам: цвет, текстура или же интенсивность. Соседние области отличаются друг от друга перечисленными выше характеристиками. Различные подходы нахождения границ между областями основываются на неоднородностях значений интенсивности яркости. Таким образом, выбор метода сегментации изображения находится в зависимости от трудности, которую нужно решить.

Алгоритмы сегментации классифицируются следующим образом:

1. Пороговые алгоритмы;
2. Алгоритмы разрастание регионов;
3. Граничные алгоритмы;
4. Сегментация на основе кластеризации.

Пороговые алгоритмы устанавливают некоторые пороговые ограничения. После преобразования в матрице изображения элементы изображения, имеющие степень яркости более порогового, будут иметь значение 1, меньшее порогового значение 0. Наилучшее значение порога формируется на основе анализа гистограмм. Далее применяется мультипороговая сегментация [24].

Пороговые алгоритмы дают возможность сегментировать обычные изображения, но не дают хорошего результата на изображениях с присутствием неравномерного освещения, вспышек и разнообразных помех. Существуют алгоритмы для устранения воздействия перечисленных

дефектов, которые подвергают анализу взвешенные значения экстремумов, а точнее анализируют интенсивность и градиент [25].

Для корректного применения алгоритма, нужно избегать «отклонения» при выборе порогового значения:

1. Нужно жестко контролировать идентичность распределения в светлых и темных областях гистограммы яркости.

2. Нужно разбивать изображение на маленькие составляющие, чтобы гистограмма яркости имела яркое выражение экстремумов.

3. Элементы должны быть довольно большими. Объем статистической выборки должен позволять удовлетворительно оценивать положения экстремумов и описывать окрестность.

Метод Оцу [24] применяется для вычисления порога бинаризации для полутонового изображения. В обычной форме алгоритм возвращает один порог интенсивности, который разграничивает пиксели на два класса: полезные и фоновые. Рассчитывая подобный порог чтобы внутриклассовая дисперсия интенсивности была минимальной. Метод Оцу является одномерным дискретным аналогом дискриминантного анализа Фишера, связан с методом оптимизации Дженкса и эквивалентен глобально оптимальному k -среднему, выполненному на гистограмме интенсивности. Расширение до многоуровневого порогового значения описано в статье [25], развитие идей описано в работах [26, 27].

Алгоритм по методу Оцу

Пусть дано монохромное изображение

$$G(i, j), i = \overline{1, Height}, j = \overline{1, Width}.$$

1. Установить счетчик повторений $k = 0$.
2. Вычислить гистограмму $p(l)$ изображения и частоту $N(l)$ для каждого уровня интенсивности изображения G .
3. Вычислить начальные значения для $\omega_1(0)$, $\omega_2(0)$, и $\mu_1(0)$, $\mu_2(0)$.

4. Для каждого значения $t = \overline{1, \max(G)}$ – полутона – горизонтальная ось гистограммы:

4.1. Обновляем ω_1, ω_2 , и μ_1, μ_2

4.2. Вычисляем $\sigma_b^2(t) = \omega_1(t), \omega_2(t)[\mu_1(t), \mu_2(t)]^2$.

4.3. Если $\sigma_b^2(t)$ больше, чем имеющееся, то запоминаем σ_b^2 и значение порога t .

4.4. Искомый порог соответствует максимуму $\sigma_b^2(t)$.

$$N_T = \sum_{i=0}^{\max(G)} p(i), \quad (3)$$

$$\omega_1(t) = \frac{\sum_{i=0}^{t-1} p(i)}{N_T} = \sum_{i=0}^{t-1} N(i), \quad \omega_2(t) = 1 - \omega_1(t). \quad (4)$$

$$\mu_T = \frac{\sum_{i=0}^{\max(G)} i \cdot p(i)}{N_T} = \sum_{i=0}^{\max(G)} i \cdot N(i), \quad (5)$$

$$\mu_1(t) = \frac{\sum_{i=0}^{t-1} i \cdot p(i)}{N_T \cdot \omega_1(t)} = \frac{\sum_{i=0}^{t-1} i \cdot N(i)}{\omega_1(t)}, \quad \mu_2(t) = \frac{\mu_T - \mu_1(t) \cdot \omega_1(t)}{\omega_2(t)} \quad (6)$$



Рисунок 1.8– Пример бинаризации с порогом по Оцу

Алгоритмы разрастания регионов эффективно применяются на изображениях, имеющих области с устойчивой связностью внутри отдельных сегментов. В результате работы таких алгоритмов происходит группирование

соседних элементов с схожими или же близкими уровнями яркости, которые в результате соединяются в однородные области [28].

К недостаткам алгоритмов наращивания областей в первую очередь относят то, собственно, что они выделяют общие фрагменты, зачастую не показывая информацию об изменении яркостей внутри областей и вероятных границах. Впрочем, эти алгоритмы показали себя более эффективными при работе с зашумленными изображениями.

Граничные алгоритмы — это алгоритмы, заключающиеся в нахождении точек, лежащих на границах областей [30]. Значения яркостей пикселя являются основными для определения границ. К часто применяемым характеристикам ещё относятся текстура и градиент.

Границы объектов на изображении в значимой степени сокращают количество данных, которые нужно обработать, и в то же время сохраняет важную информацию об объектах на изображении, их форму, величину, количество.

Алгоритмы сегментации имеют конкретные области применения, которые зависят от:

- входных данных,
- требований к распознаванию,
- объема вычисления,
- скорости.

При выборе алгоритма сегментации были рассмотрены наиболее популярные и известные методы. К ним относятся оператор Робертса, Собеля, детектор границ Кэнни.

Оператор Собеля –оператор вычисляет приближённое значение градиента яркости изображения. В результате применения оператора Собеля получается изображение, каждая точка которого будет являться либо вектор градиента яркости или норма в этой точке [29].



Рисунок 1.9 – Результат применения оператор Собеля

Как видно на рисунке 1.9, оператор выделил границы. Стоит отметить, что на области автомобиля явно виден контур автомобильного номера и его символы.

Оператор Роберта выполняет простые и быстрые вычисления двумерного пространственного измерения на изображении. Данный метод подчеркивает области высокой пространственной частоты, которые зачастую соответствуют краям. На вход оператора подается полутоновое изображение. Значение пикселей выходного изображения в каждой точке подразумевает некоторую значение пространственного градиента входного изображения в данной же точке. Обнаружение пределов границ данным методом вычислительно гораздо проще, чем методом Собеля, но приводит к частому ложному срабатыванию фильтра в точечных всплесках яркости, собственно, что приводит к зашумлению результирующего изображения.

Детектор границ Кэнни является одной из самых популярных алгоритмов обнаружения контуров. Важным шагом в данном алгоритме является устранение шума на контурах, который в значительной мере может

повлиять на результат, при этом необходимо максимально сохранить границы. Для этого необходим тщательный подбор порогового значения при обработке [34].

Сегментация на основе кластеризации. Преимуществом алгоритмов сегментации на основе кластеризации состоит в автоматизации процесса поиска величины параметров для разделения классов. Существуют множество алгоритмов кластеризации, среди которых можно выделить метод Виолы-Джонса, CURE, BIRCH [31-32], k-средних [34], метод опорных векторов [25]. Указанные алгоритмы находят кластеры, соответствующие определенной статической модели.

Выделим также подход для обработки изображений с использованием **сверточных нейронных сетей** [15]. В сверточной нейронной сети происходит чередование двух типов слоев: сверточных слоев и слоев подвыборки. К преимуществам можно отнести.

1.5 Библиотеки для обработки изображений

OpenCV – это библиотека программных функций, направленных в основном на компьютерное зрение в реальном времени. Первоначально разработана Intel, позже поддерживалась WillowGarage, а затем Itseez. Библиотека является кроссплатформенной и бесплатной для использования подлицензией BSD с открытым исходным кодом. OpenCV поддерживает фреймворки глубокого обучения – TensorFlow, Torch / PyTorch и Caffe [18].

EmguCV — кроссплатформенная «обёртка» для .NET библиотеки обработки изображений OpenCV (основной библиотеки), EmguCV также называют библиотекой машинного зрения [19].

Для обработки изображений мы использовали библиотеку компьютерного зрения EmguCV, которая позволяет вызывать методы OpenCV в .NET-совместимых языках такими как C #, VB, VC ++, IronPython и т.д. Может использоваться для разработки мобильных приложений, в том числе тех, которые используют VisualStudio, Xamarin. Таким образом проект

может быть реализована для всех популярных на сегодняшний день мобильных платформ.

1.6 Выводы по литературному обзору

Для предварительной обработки изображений чаще всего используются: фильтр Гаусса – в основном для удаления шумов, детектор Кэнни – для поиска контуров объектов, детектор Хаара – для поиска объекта на изображении, гистограммы яркости – для поиска границ объектов. Анализ существующих на Российском рынке программных продуктов показал, что заявленная точность распознавания обычно завышена и не совпадает с реальной, при испытании демо-версий надежно распознаются лишь чистые номера высокой контрастности и относительно большого разрешения поэтому нами поставлена цель улучшить процент распознавания автомобильного номерного знака.

На основании литературного обзора можно сделать вывод, что разработка алгоритма распознавания номерных знаков на полученном локализованном изображении автотранспортных средств и разработка программы является актуальной задачей.

2 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМА

На основе проведенного литературного обзора разработан алгоритм распознавания автомобильного регистрационного номерного знака на локализованной области. Было решено разработать настольное программное обеспечение с удобным интерфейсом для реализации алгоритма распознавания автомобильного регистрационного номерного знака на локализованной области.

Программное обеспечение, реализующее алгоритм разработано на языке программирования C# с использованием библиотек EmguCV и AvgSdv. Для предобработки изображения используется библиотека EmguCV, а для распознавания символов – AvgSdv [35].

2.1 Архитектура программного обеспечения

Разрабатываемое программное обеспечение содержит четыре компонента – компонент взаимодействия с пользователем и вывода результатов распознавания, компонент распознавания номерных знаков, компонент предобработки изображения, компонент распознавания символов. (рисунок 2.1).

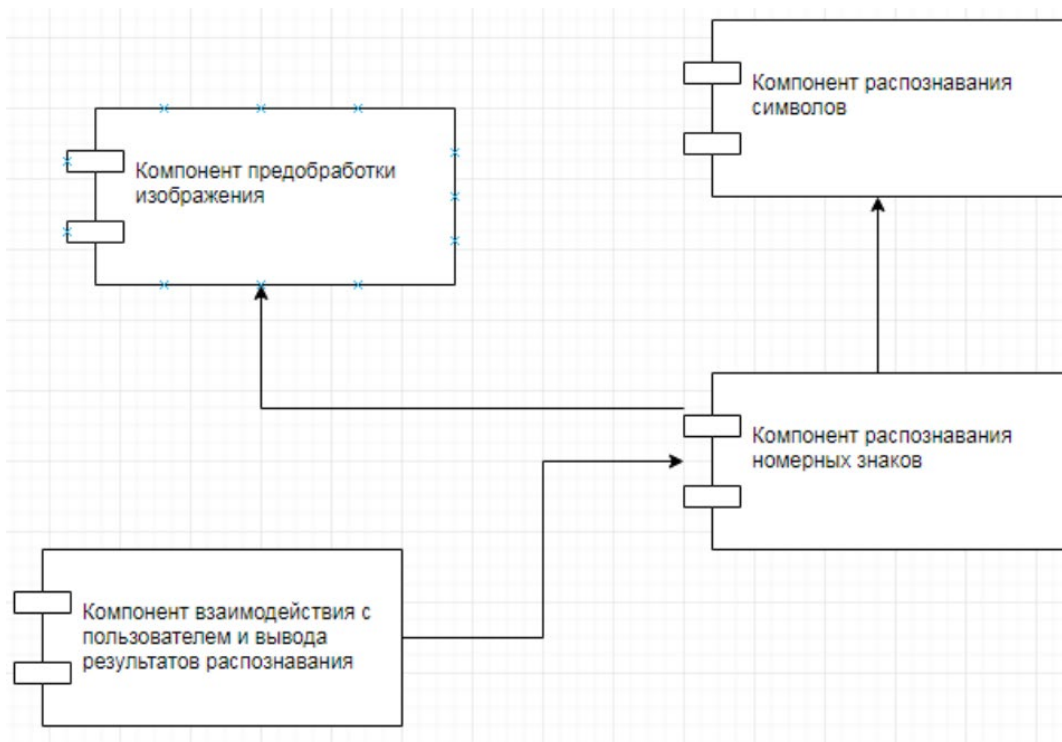


Рисунок 2.1 – Диаграмма компонентов программного обеспечения

Компонент взаимодействия с пользователем и вывода результатов распознавания позволяет пользователю при запуске программного обеспечения выбрать изображение номерной пластины, в котором необходимо распознать автомобильный номер и после обработки вывести результат распознавания.

Компонент распознавания номерных знаков позволяет распознать автомобильный номер на изображении, которого выбрал пользователь после запуска приложения.

Компонент предобработки изображения переводит изображение в градации серого, удаляет шумы, производит бинаризацию и другие операции для предварительной обработки.

Компонент распознавания символов позволяет распознать символы номерного знака, полученные путем сегментирования.

2.2 Выделение номерной рамки

Допустим, что имеется изображение автомобиля. На первом этапе работы необходимо выделить номерную рамку (рисунок 2.2). Для выделения рамки мы будем использовать примитивы Хаара из библиотеки OpenCV.



Рисунок 2.2 – Выделенная номерная рамка с использованием примитивов Хаара из библиотеки OpenCV

Далее к номерной рамке применим алгоритм нормализации угла наклона и масштаба.



Рисунок 2.3 – Исходное изображение



Рисунок 2.4 – Нормализация угла наклона

Допустим, что номер может иметь диапазон поворота от -10° до $+10^\circ$. При этом будет происходить обработка каждый раз нового кадра с шагом в 0.1° . Обработка каждого кадра происходит независимо от предыдущих. Какая гипотеза по повороту даст наилучший итог, та и победит.

Нормализация угла наклона приведено на рисунке 2.4. Для каждого кадра рассчитывается нижняя граница изображения. После работы алгоритма выигрывает угол, для которого была рассчитана самая верхняя граница. Это и станет разыскиваемый угол. Масштаб изображения (по аналогии с поворотом) имеет диапазон от 1 до 3 с шагом 0.1.

После того как найден угол наклона автомобильного номера, необходимо найти его границы. Поиск нижней границы автомобильного номера строится на анализе гистограммы яркости, границей поиска служит зеленая линия (рисунок 2.5).



Рисунок 2.5 – Поиск нижней границы номерного знака

Во время тестов опытным путём установлено, что гипотеза с применением гистограммы яркости для поиска верхней границы, автомобильных номеров работает в 50% случаев, собственно, что недопустимо для разрабатываемой системы. В связи с этим было решено научить каскад Хаарана каждую букву и проанализировать верхние грани букв, тем самым вероятно будет получить верхнюю границу автомобильного номера.

В некоторых случаях, алгоритм с использованием каскада Хаара может не дать результата, это свойственно для изображений с довольно невысоким разрешением. Для этих изображений используется, как альтернативный алгоритм, поиск границы с применением гистограммы яркости.

2.3 Решение задачи распознавания автомобильного номера

После выделения рамки автомобильного номера процесс решения задачи распознавания автомобильных номерных знаков в общем виде может быть представлен в последовательности шагов, показанных на рисунке 2.6.



Рисунок 2.6 – Общая схема решения задачи распознавания автомобильного номерного знака

Подробная структурная схема процесса представлена на рисунке 2.7.

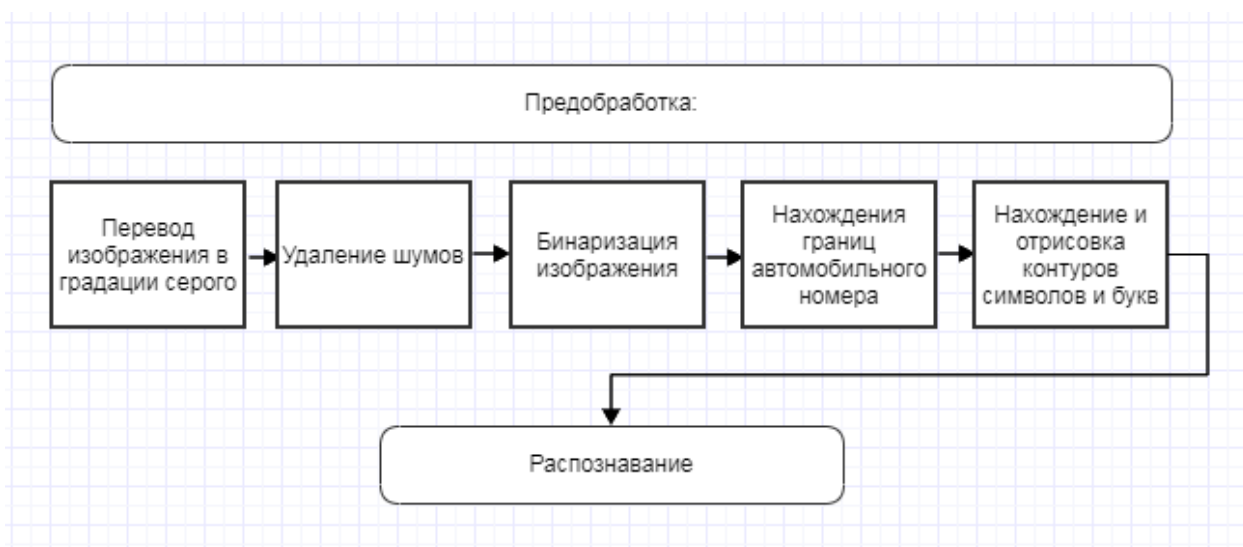


Рисунок 2.7 – Структурная схема процесса распознавания

Допустим, что у нас имеется локализованная рамка с регистрационным номером. Предлагаемый алгоритм распознавания номерных знаков представляется в виде последовательности следующих шагов:

Шаг 1. Перевод изображения автомобильного номера в градации серого.

Шаг 2. Удаление шумов.

Шаг 3. Бинаризация изображения.

Шаг 4. Нахождение границ автомобильного номера.

Шаг 5. Нахождение и отрисовка контуров символов и букв.

Шаг 6. Сегментация номерной пластины

Шаг 7. Распознавание символов и букв.

Далее приведем описание указанных выше шагов алгоритма. Для преобразование в градации серого мы использовали **оператор Собеля** (рисунок 2.8).



Рисунок 2.8 – Преобразование изображения в градации серого.

Оператор Собеля

Для избавления от посторонних шумов на изображении мы использовали **фильтр Гаусса** и применили **метод Оцу** для бинаризации (рисунок 2.9).



Рисунок 2.9 – Бинаризация с порогом по Оцу

Для поиска контуров мы использовали детектор границ Кэнни. При вычислительных экспериментах мы также рассмотрели фильтр Собеля, однако качество его работы оказалось хуже детектора Кэнни.

На рисунке 2.10 представлен результат сегментации номерного знака на основе выбранного метода контурного анализа.

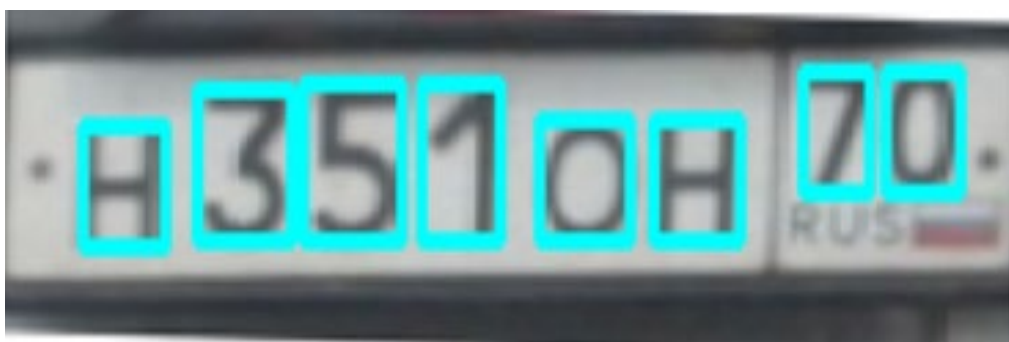


Рисунок 2.10 - Сегментация символов номера

Полученные координаты прямоугольных областей, обозначенные бирюзовым цветом, позволяют сегментировать номерную пластину, вырезав каждый сегмент для последующего распознавания. На рисунке 2.11 показан результат извлечения сегментов из изображения номерной пластины.



Рисунок 2.11 – Результат извлечения сегментов

Каждый извлечённый сегмент приводится к размеру 20×20 пикселей и сохраняется во временную папку для дальнейшего распознавания. Сегменты нумеруются слева направо от 1 до 8 (или 9, в случае, когда код региона трехзначный).

2.4 Графический интерфейс пользователя

При запуске программного обеспечения открывается удобный интерфейс программы, на правом верхнем углу находится список с загруженным исходным изображением из папки image, необходимо выбрать изображение для обработки и распознавание номерного знака, после чего будет запущена обработка и распознавание изображения. В данном программном обеспечении на вход подается изображение, содержащее автомобиль (рисунок 2.12).

Значения	Символ 1	Символ 2	Символ 3	Символ 4	Символ 5	Символ 6	Символ 7	Символ 8	Символ 9
0	26,3	38,3	42,9	38,3	92,1	28,8	23,5	54,3	
1	15,1	17,2	28,2	93,2	23,8	19,2	15,8	16,1	
2	21,7	38,8	42,6	29,1	28,2	26,7	48,2	22,4	
3	19,6	73,7	53,5	33,3	25,0	24,3	36,4	17,8	
4	1,5	3,7	12,7	27,5	8,8	7,1	6,6	6,1	
5	26,6	54,7	87,5	40,2	31,3	36,7	28,9	24,3	
6	18,2	29,7	55,2	29,4	39,8	23,9	14,2	28,9	
7	0,4	24,6	22,6	14,7	4,5	5,8	64,8	4,2	
8	33,4	49,0	58,3	36,5	47,1	37,0	26,9	52,3	
9	17,3	23,0	37,5	32,6	28,6	19,7	28,9	30,9	
A	0,7	11,9	17,0	26,3	2,4	9,8	12,4	6,0	
B	53,7	45,2	55,8	29,2	54,4	54,7	23,1	36,3	
E	20,5	33,1	56,4	24,9	20,1	30,6	27,0	22,1	
K	31,8	20,8	17,6	29,5	6,7	28,1	18,2	7,3	
M	39,0	16,9	17,7	36,7	20,0	44,8	14,1	8,5	
H	81,1	30,3	32,8	51,7	28,3	80,2	23,1	22,3	
O	17,2	26,2	28,5	28,2	66,3	19,0	13,8	43,1	
P	30,7	22,3	25,6	20,1	19,9	32,3	22,7	18,5	
C	4,6	19,8	34,9	26,7	26,2	13,7	16,7	20,5	
T	9,9	13,6	14,5	3,0	2,9	4,5	30,4	6,8	
X	4,5	18,8	18,7	20,8	1,3	10,6	26,8	0,3	
Y	9,0	20,3	27,4	20,2	6,4	15,0	42,7	6,6	

Рисунок 2.12 – Результат работы программы

На рисунке 2.12 приведен графический интерфейс разработанной программы. Главное окно программы функционально разделено на три части: а) выбор исходного изображения, б) служебная информация и в) результат распознавания. К исходному изображению – серый автомобиль с

государственным номером Н 351 ОН 70 (рис. 2.12, в центре) – применяем оператор Собеля по горизонтали и вертикали и получаем черно-белое изображение (рис. 2.12, слева), далее вычисляем для каждой горизонтали разность яркостей соседних пикселей (оранжевая линия) и находим среднее значение всех яркостей по горизонтали (вертикальная бирюзовая линия), затем находим локальные максимумы для кривой выше средней (желтые маркеры). После этого вырезаем горизонтальную область изображения, в которой располагается глобальный экстремум и повторно применяем оператор Собеля. Далее находим по вертикали яркостную кривую, затем по прямоугольникам локализованной области определяем являются ли символы автомобильным номером (прямая зеленого цвета). Окончательно бинаризуем локализованную область и с помощью функции AvgSdv из библиотеки компьютерного зрения Emgu определяем статистические характеристики (среднее, стандартное отклонение) для каждого элемента изображения (символа) и сравниваем с эталоном каждой буквы и цифры. Результаты работы программы – государственный номер Н351ОН70 – выводится в нижней части главного окна. Справочно выводится таблица, в которой отображены ошибка посимвольного распознавания каждой буквы и цифры. В таблице выделены ячейки, из которых определен номер.

3 РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

3.1 Тестирование программного продукта

Сегодня тестирование является неотъемлемой частью процесса производства программных продуктов. Качественное тестирование помогает своевременно выявлять и исправлять ошибки, тем самым уменьшая риски и затраты на разработку программного обеспечения.

Следующая серия вычислительных экспериментов была проделана с целью тестирования среднего времени обработки номерного знака в зависимости от расстояния до объекта. Результаты вычислений показали, что с увеличением расстояния от 1 до 10 м время обработки возрастает с 512 до 817 мс (прирост на 59%).

3.2 Тестирование алгоритма распознавания

Для выявления точности распознавания было подано на вход программному обеспечению 5 486 изображений.

По полученным результатам алгоритма распознавания, получены следующие результаты

Таблица 3.1 – Тестирование алгоритма

Общее количество поданных номерных знаков	X=	5486
Количество правильно распознанных номерных знаков	Y=	5057

Средняя точность распознавания алгоритма

$$\frac{\text{количество верно распознанных номеров}}{\text{объем выборки}} \cdot 100\% = \frac{5057}{5486} \cdot 100\% = 92,2\%.$$

Среднее время, затраченное на распознавание автомобильного номера – 550 мс. Сравнения программного обеспечения по точности распознавания представлено в таблице 3.2. Как видно из таблицы предлагаемый алгоритм на тестовой выборке уступает только одному из рассмотренных аналогов (выделено жирным шрифтом).

Таблица 3.2 – Сравнение точности программного обеспечения

Наименование системы	Точность распознавания, %
Авто-Инспектор	94,19
Разработанный алгоритм	92,20
АвтоУраган	90,45
AutoTrassir	87,07
Поток	86,38
CVS Авто	82,09
ИнтеграВидеоАвто	81,77
MegaCar	81,06
OverseerTraffic	79,89
ТелеВизардАвто	51,33

4 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

Поддержание комфортных условий труда и отдыха ведет к повышению производительности труда. Анатомические и физиологические особенности человека должны приниматься во внимание при выборе оптимальных параметров среды. Поддерживание безопасности на работе и отдыхе способствует сохранению жизни и здоровья человека в результате уменьшения, как числа несчастных случаев, так и темпов роста болезней под воздействием природы и экологическими катастрофами, и чрезвычайными обстоятельствами.

Раздел, посвященный социальной ответственности организации, включает в себя следующие составляющие: правовые и организационные вопросы обеспечения безопасности; производственная безопасность; экологическая безопасность и безопасность в чрезвычайных ситуациях.

Научно-исследовательский проект представляет собой разработку программного продукта и предполагает большой объем работы с ПК, поэтому важным критерием безопасности является организация рабочего места и режима трудовой деятельности.

4.1 Правовые и организационные вопросы обеспечения безопасности

4.1.1 Правовые нормы трудового законодательства для рабочей зоны

Законодательство РФ об охране труда основывается на Конституции РФ и состоит из федерального закона, других федеральных законов и иных нормативных правовых актов субъектов РФ. Среди них можно выделить федеральный закон “Об обязательном социальном страховании от несчастных случаев на производстве и профессиональных заболеваний”. Для реализации этих законов приняты Постановления Правительства РФ “О государственном надзоре и контроле за соблюдением законодательства РФ о

труде и охране труда”, “О службе охраны труда”, “О Федеральной инспекции труда” и др. [37]

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.

Выпускная квалификационная работа студента выполнялась в десятом корпусе ТПУ на кафедре информационных систем и технологий. Рабочее место находится на четвертом этаже здания и представляет собой комнату длиной – 9 м., шириной – 5 м. и высотой – 3 м. Естественное освещение кабинета осуществляется посредством двумя окнами размерами 2,2 м. х 1,5 м. Дверь – деревянная одностворчатая, белого цвета. Высота двери – 2 м., ширина – 1 м. Стены комнаты окрашены водоэмульсионной краской бежевого цвета. Потолок подвесной, плиточный. Пол покрыт линолеумом. Площадь кабинета составляет 45 м², объем – 135 м³. Согласно СанПиН 2.2.2/2.4.1340-03 [36], норма площади рабочего места с персональным компьютером составляет 4,5 м². В рассматриваемой аудитории установлено 10 рабочих мест с персональными компьютерами и жидкокристаллическими экранами. Соответственно, на одного человека приходится 4,5 м², что соответствует вышеуказанным требованиям

4.1.2 Производственная безопасность

Для обеспечения производственной безопасности необходимо проанализировать воздействия на человека вредных и опасных производственных факторов, которые могут возникать при разработке или эксплуатации проекта.

Все производственные факторы классифицируются по группам элементов: физические, химические, биологические и психофизические. Для данной работы целесообразно рассмотреть физические и психофизические вредные и опасные факторы производства, характерные как для рабочей зоны программиста, как разработчика рассматриваемой в данной работе

системы, так и для рабочей зоны пользователя готового продукта – аналитика-оператора ПЭВМ. Выявленные факторы представлены в таблице 4.1.

Таблица 4.1 - Возможные опасные и вредные факторы

№	Факторы (ГОСТ 12.0.003-2015)	Этапы работы			Нормативные документы
		Разрабо тка	Изготов ление	эксплуат ация	
1	Отклонение показателей микроклимата	+	+	+	СанПиН 2.2.4.548-96. Гигиенические требования к микроклимату производственных помещений; СП 52.13330.2011 Защита от шума. СП 52.13330.2016 Естественное и искусственное освещение. СанПиН 2.2.4.3359-16. Санитарно-эпидемиологические требования к физическим факторам на рабочих местах.
2	Превышение уровня шума	+	+	+	
3	Недостаточная освещенность рабочей зоны	+	+	+	
4	Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека		+	+	
5	Повышенный уровень электромагнитных излучений	+	+	+	

4.2 Анализ вредных и опасных факторов, которые может создать объект исследования.

4.2.1 Отклонение показателей микроклимата

Одним из необходимых условий здорового и высокопроизводительного труда является обеспечение нормальных микроклиматических условий, являющихся важной характеристикой санитарно-гигиенических условий труда. К микроклиматическим условиям на рабочем месте относятся температура воздуха, относительная влажность, скорость движения воздуха. Микроклиматические условия в помещении очень часто изменчивы и зависят от метеорологических условий наружной атмосферы, мощности источников тепловыделений в помещении, расположения рабочего места, воздухообмена и т.д. Вызываемое метеорологическими условиями тепловое воздействие или

воздействие холода может привести к значительным изменениям жизнедеятельности организма человека и вследствие этого к снижению производительности труда, повышению общей заболеваемости работающих. Для исключения вредного влияния микроклиматических факторов на организм человека и создания нормальных условий труда в рабочей зоне параметры воздушной среды должны соответствовать СанПин 2.2.4.548-96. [40]

В производственных помещениях для работы с ПЭВМ происходит постоянное выделение тепла самой вычислительной техникой, вспомогательными приборами и средствами освещения. Поскольку оператор расположен в непосредственной близости с источниками выделения тепла, то данный фактор является одним из важнейших вредных факторов производственной среды оператора ПЭВМ, а высокая температура воздуха способствует быстрому перегреву организма и быстрой утомляемости.

Санитарные нормы устанавливают оптимальные и допустимые значения величин показателей микроклимата рабочих мест для различных категорий работ в теплый и холодный периоды года. Для программиста или оператора ПЭВМ категория работ является лёгкой (1а), т.к. работа проводится сидя, без систематических физических нагрузок. Оптимальные параметры микроклимата в офисных помещениях приведены в таблице 4.2.

Таблица 4.2 – Оптимальные параметры микроклимата производственных помещений оператора ПЭВМ.

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность, %	Скорость движения воздуха, м/с
Холодный	22-24	21-25	60-40	0,1
Теплый	23-25	22-26	60-40	0,1

Для обеспечения нормальных метеорологических условий и поддержания теплового равновесия между телом человека и окружающей средой в лаборатории проводится ряд мероприятий, основными из которых являются: обеспечение приточно-вытяжной вентиляцией, системой водяного

отопления, которые всегда должны находиться в рабочем состоянии. В нашем помещении существуют действующие вентиляция и водяное отопление, которые позволяют поддерживать необходимую температуру, соответствующую СанПин 2.2.4.548-96. [40]

4.2.2 Повышенный уровень шума

Шум в большей или меньшей степени может временно активизировать или постоянно подавлять определенные психические процессы организма человека. Физиопатологические последствия могут проявиться в форме нарушения слуха и других анализаторов.

Шум не только ухудшает самочувствие человека и снижают производительность труда в среднем на 10-15%, но и очень часто приводят к профессиональным заболеваниям.

Шум создаются работающими приборами, вентиляторами. В большинстве случаев технически трудно снизить шум до очень малых уровней, поэтому при нормировании исходят не из оптимальных, а из терпимых условий, т.е. таких, когда вредное действие шума на человека не проявляется или проявляется незначительно.

Нормирование допустимых уровней звукового давления производится в соответствии с ГОСТ 12.1.003.-2014 [39]. По данному ГОСТу уровень звука в производственных помещениях не должен превышать 85 дБА [39].

Так как, выполнение выпускная квалификационная работа происходило в аудитории 427а 10-го корпуса НИИ ТПУ, то можно рассмотреть такие источники шума, как:

- шум работающих в помещении приборов (персональный компьютер, принтер, вентиляторы);
- посторонние звуки за пределами помещения (но в пределах 10-го корпуса НИИ ТПУ);
- звуки улицы (транспорт, различные производственные шумы).

Работающие люди в таких зонах должны быть снабжены предметами индивидуальной защиты. Согласно ГОСТу к средствам индивидуальной защиты от повышенного уровня шума относятся:

- противозумные шлемы;
- противозумные вкладыши (беруши);
- противозумные наушники.

4.2.3 Недостаточная освещенность рабочей зоны

Освещение оказывает большое влияние на зрительную работоспособность, физическое и моральное состояние людей, производительность и качество труда. Следовательно, для сохранения работоспособности в течение рабочей смены необходимым условием является рациональное освещение помещений и рабочих мест, которое обеспечивается естественным и искусственным освещением согласно СП 52.13330.2016 [38].

Искусственное освещение в помещениях эксплуатации компьютеров должно осуществляться системой общего равномерного освещения.

Для обеспечения нормативных значений освещенности в помещениях следует проводить чистку стекол оконных проемов и светильников не реже двух раз в год и проводить своевременную замену перегоревших ламп.

В качестве искусственного освещения выбираем люминесцентные открытые двухламповые светильники типа ОДО-2-40. Расчёт общего равномерного искусственного освещения горизонтальной рабочей поверхности выполняется методом коэффициента светового потока, учитывающим световой поток, отражённый от потолка и стен. Длина помещения $B=9$ м, ширина $A=5$ м; высота $H=3$ м.

Минимальное освещение $E_{т}=150$ лк, в соответствии с разрядом зрительной работы.

Коэффициент запаса для помещения с малым выделением пыли при люминесцентных лампах: $k=1,5$.

Расчетная освещенность:

$$E_p = E_T * k = 150 * 1,5 = 225 \text{ лк.}$$

Свес светильников с потолка: $h_c = 0,2 \text{ м.}$

Высота рабочей поверхности над полом: $h_p = 0,8 \text{ м.}$

Высота подвеса светильников над рабочим местом:

$$h = H - h_c - h_p = 3 - 0,2 - 0,8 = 2 \text{ м.}$$

Площадь помещения, где A – длина, м; B – ширина, м.

$$S = A * B = 9 * 5 = 40 \text{ м}^2.$$

Удельная мощность: $w = 16 \text{ Вт/м}^2.$

Суммарная установленная мощность:

$$P = S * w = 40 * 16 = 640 \text{ Вт.}$$

Мощность одного светильника: $P_i = 80 \text{ Вт.}$

Количество светильников:

$$n = P / P_i = 640 / 80 = 8 \text{ шт.}$$

Для равномерного общего освещения светильники располагаются в коридорном порядке.

Размеры светильника типа ОДОР-2-40: Этот светильник имеет две лампы мощностью 40 Вт каждая, длина светильника равна 1227 мм, ширина – 265 мм.

Величина светового потока лампы:

$$F_{\text{п}} = (E * k * S * z) / (n * p),$$

где $E = 150 \text{ лк}$ - минимальная освещенность, $k = 1,5$ - коэффициент запаса для люминесцентных ламп, $S = 40 \text{ м}^2$ - площадь помещения $z = 0,9$ - коэффициент неравномерности освещения.

Индекс помещения:

$$i = S / (h * (A + B)) = 40 / (2 * (9 + 5)) = 0,95$$

Коэффициент использования светового потока, показывающий какая часть светового потока ламп попадает на рабочую поверхность, для светильников типа ОДОР с люминесцентными лампами при $\rho_{\text{п}} = 70\%$, $\rho_{\text{с}} = 50\%$

и индексе помещения $i=0,95$, коэффициент использования светового потока равен $p=0,48$.

$$F_{\Pi} = (150 \cdot 1,5 \cdot 40 \cdot 0,9) / (8 \cdot 0,48) = 8100 / 3,84 = 2109 \text{ лм}$$

Согласно вычислениям, потребный световой поток равен 2109 лм.

Делаем проверку выполнения условия:

$$-10\% \leq \frac{F_{\text{Лб}} - F_{\Pi}}{F_{\text{Лб}}} \cdot 100\% \leq 20\%$$

$$\frac{F_{\text{Лб}} - F_{\Pi}}{F_{\text{Лб}}} \cdot 100\% = \frac{2600 - 2109}{2109} \cdot 100\% = 18,9\%$$

Таким образом: $-10\% \leq 18,9\% \leq 20\%$, необходимый световой поток светильника не выходит за пределы требуемого диапазона.

4.2.4 Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека

Электрический ток представляет собой опасность, которая не предупреждает о своем присутствии (нет видимых движущихся частей, свечения, запаха, шума), а в случае повреждения электроустановок (принтер, персональный компьютер, проектор, сканер), вокруг места повреждения возникает опасное электрическое поле. Особенно опасно прикосновение человека к токоведущим частям находящимся под напряжением. Вследствие теплового воздействия электрического тока при непосредственном прикосновении человека к токоведущим частям и при воздействии электрической дуги возникают внешние местные поражения (ожоги). Ожоги могут быть поверхностные или глубокие, сопровождающиеся поражением не только кожных покровов, но и подкожной ткани, жира, глуболежащих мышц и кости. Различают три степени электрических ожогов:

1. Покраснение кожи.
2. Образование пузырей.
3. Обугливание и омертвление кожи.

Раны от ожогов заживают очень долго, а поражение 2/3 поверхности тела может привести к смертельному исходу. Возникают также механические повреждения — разрыв тканей и некоторых внутренних органов, это может быть следствием динамических перенапряжений при прохождении через тело человека электрического тока (практически — тока короткого замыкания). Наибольшую опасность при всех видах поражения представляет электроудар, когда при прохождении тока через тело человека поражается весь организм в целом, возникают судороги, расстройство дыхания, аритмия работы сердца. Степень опасности силы тока зависит от силы тока, проходящего через организм. Если сила тока, проходящего через организм 1,5 мА (при постоянном напряжении), то в месте контакта с токоведущими частями ощущается зуд и нагрев. Такую силу тока называют порогом ощущения. Увеличение силы тока до 10 мА при переменном и до 50 мА при постоянном напряжении вызывает у человека сильные боли в пальцах и кистях рук. При такой силе тока человек еще может самостоятельно оторваться от токоведущих частей. Такую силу тока называют условно безопасной. Дальнейшее увеличение от 10 мА до 100 мА при переменном и от 50 мА до 100 мА при постоянном напряжении вызывает очень сильные боли, руки парализуются, наступает паралич дыхания, самостоятельно оторваться от токоведущих частей невозможно. Сила тока при снижении сопротивления человеческого тела постоянно возрастает и при достижении 100 мА, как при переменном, так и при постоянном напряжении наступает клиническая смерть (отсутствие внешних признаков жизни).

Опасное и вредное воздействие на людей электрического тока, электрической дуги и электромагнитных полей проявляются в виде электротравм и профессиональных заболеваний. Для предотвращения этих и других несчастных случаев необходимо разработать электробезопасность конструкций. Согласно ПЭУ, данная помещения без повышенной электроопасности, которое характеризуется отсутствием условий, создающих

повышенную и или особую опасность. Основные способы и средства электрозащиты:

- изоляция токопроводящих частей и ее непрерывный контроль;
- установка оградительных устройств;
- предупредительная сигнализация и блокировки;
- использование знаков безопасности и предупреждающих плакатов;
- использование малых напряжений;
- электрическое разделение сетей;
- защитное заземление;
- выравнивание потенциалов;
- зануление;
- защитное отключение;
- средства индивидуальной электрозащиты.

4.2.5 Повышенный уровень электромагнитных излучений

Уровень электромагнитных излучений на рабочем месте оператора ПЭВМ является вредным фактором производственной среды, величины параметров которого определяются СанПиН 2.2.2/2.4.1340-03 [36]. Основными источниками электромагнитных излучений в помещениях для работы операторов ПЭВМ являются дисплеи компьютеров и мобильных устройств, сеть электропроводки, системный блок, устройства бесперебойного питания, блоки питания.

Излучения, применительно к дисплеям современных ПЭВМ, можно разделить на следующие классы:

- переменные электрические поля (5 Гц – 400 кГц);
- переменные магнитные поля (5 Гц – 400 кГц).

Воздействие данных излучений на организм человека носит необратимый характер и зависит от напряженности полей, потока энергии, частоты колебаний, размера облучаемого тела. При воздействии полей,

имеющих напряженность выше предельно допустимого уровня, развиваются нарушения нервной системы, кровеносной сердечно-сосудистой системы, органов пищеварения и половой системы. [41] В таблице 4.3 приведены допустимые уровни параметров электромагнитных полей.

Таблица 4.3 – Временные допустимые уровни электромагнитных полей, создаваемых ПЭВМ на рабочих местах.

	Наименование параметров	Допустимые значения
Напряженность электрического поля	В диапазоне частот 5 Гц – 2 кГц	25 В/м
	В диапазоне частот 2 кГц – 400 кГц	2,5 В/м
Плотность магнитного потока	В диапазоне частот 5 Гц – 2 кГц	250 нТл
	В диапазоне частот 2 кГц – 400 кГц	25 нТл
Электростатический потенциал видеомонитора	экрана	15 кВ/м

Для уменьшения уровня электромагнитного поля от персонального компьютера рекомендуется включать в одну розетку не более двух компьютеров, сделать защитное заземление, подключить компьютер к розетке через нейтрализатор электрического поля.

К средствам индивидуальной защиты при работе на компьютере относят спектральные компьютерные очки для улучшения качества изображения, защиты от избыточных энергетических потоков видимого света и для профилактики «компьютерного зрительного синдрома». Очки уменьшают утомляемость глаз на 25-30 %. Их рекомендуется применять всем операторам при работе более 2 ч в день, а при нарушении зрения на 2 диоптрии и более – независимо от продолжительности работы.

4.3 Экологическая безопасность

В данном разделе рассматривается воздействие на окружающую среду деятельности по разработке проекта, а также самого продукта в результате его реализации на производстве.

Разработка программного обеспечения и работа за ПЭВМ не являются экологически опасными работами, потому объект, на котором производилась разработка продукта, а также объекты, на которых будет производиться его использование операторами ПЭВМ относятся к предприятиям пятого класса, размер селитебной зоны для которых равен 50 м. [42]

Программный продукт, разработанный непосредственно в ходе выполнения магистерской диссертации, не наносит вреда окружающей среде.

Средства, необходимые для разработки и эксплуатации программного комплекса могут наносить вред окружающей среде. При рассмотрении влияния процесса утилизации персонального компьютера были выявлены особо вредные выбросы согласно ГОСТ Р 51768-2001.[43] В случае выхода из строя компьютеров, они списываются и отправляются на специальный склад, который при необходимости принимает меры по утилизации списанной техники и комплектующих. Люминесцентные лампы, применяющиеся для искусственного освещения рабочих мест, также требуют особой утилизации, т.к. в них присутствует 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. В настоящее время в Томской области утилизацией занимаются две компании: городской полигон и ООО НПП «Экотом». Утилизацией опасных бытовых отходов занимаются компании: ООО «Торем», ООО «СибМеталлГрупп».

4.4 Безопасность при чрезвычайных ситуациях

Чрезвычайные ситуации (ЧС) – совокупность таких обстоятельств, которые сопровождаются разрушениями, поражениями людей, изменением

экологической обстановки. К чрезвычайным ситуациям относятся: производственные аварии, стихийные бедствия, военные конфликты.

В условиях ЧС необходимо знать правила поведения во избежание паники и несчастных случаев.

Одними из наиболее вероятных и разрушительных видов чрезвычайных ситуаций являются пожар или взрыв на рабочем месте.

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

- пожары и взрывы в зданиях и на коммуникациях;
- внезапное обрушение зданий.

Наиболее характерной для объекта, где размещаются рабочие помещения, оборудованные ПЭВМ, чрезвычайной ситуацией является пожар.

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.

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 проводить 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

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).

Самое главное – при любой чрезвычайной ситуации сохранять спокойствие и не забывать отключить электроэнергию и электроприборы. Иначе, это приведет к еще большим разрушениям.

Выводы по разделу

Данный раздел включает в себя комплекс мер и мероприятий по обеспечению безопасных условий труда и жизни для людей, находящихся на рабочих местах в офисном помещении. Проанализированы законодательные и нормативные документы, затрагивающие данную сферу исследования. Был проведен анализ влияния вредных факторов на процесс работы сотрудников и на окружающую среду. Разработаны организационные мероприятия по обеспечению организационной безопасности и безопасности в чрезвычайных ситуациях, что позволит свести к минимуму вероятность возникновения чрезвычайных и опасных ситуаций, а также вероятность возникновения профессиональных заболеваний.

5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

5.1 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является проектирование и создание конкурентоспособных разработок, технологий, отвечающих современным требованиям в области ресурсоэффективности и ресурсосбережения.

Для достижения описанных выше целей необходимо решить следующие задачи:

- определить потенциальных потребителей путем рассмотрения целевого рынка и проведения его сегментирования;
- провести SWOT-анализ для комплексной оценки угроз, возможностей, сильных и слабых сторон научно-исследовательского проекта, а также для анализа внешней и внутренней среды проекта;
- рассчитать затраты на реализацию проекта;

5.2 Предпроектный анализ

5.2.1 Потенциальные потребители результатов исследования

Для анализа потребителей результатов исследования необходимо рассмотреть целевой рынок.

Целевой рынок – сегменты рынка, на котором будет продаваться в будущем разработка. В свою очередь, сегмент рынка – это особым образом выделенная часть рынка, группы потребителей, обладающих определенными общими признаками.

Выполненная работа направлена для разработки алгоритма и программы распознавания автомобильного регистрационного номерного знака на локализованной области.

Потенциальными потребителями результатов НИР являются отрасли (экономика, сферы услуг и т.д.), применяемые алгоритм распознавания номеров в решение своих задач.

5.2.2 Анализ конкурентных технических решений

Анализ конкурентных технических решений позволяет провести оценку сравнительной эффективности научной разработки и определить направления для ее будущего повышения. Основным методом, используемым в данной работе, является метод динамических ядер или К-средних. Конкурентные методы: Деревья решений, метод опорных векторов (SVM).

Анализ конкурентных технических решений определяется по формуле:

$$K_i = \sum_{i=1}^n B_i B_i \quad (1)$$

где K – конкурентоспособность разработки или конкурента; B_i – вес -го показателя (в долях единицы); B_i – балл i -го показателя.

Таблица 5.1 – Оценочная карта для сравнения конкурентных технических решений

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Бф	Бк1	Бк2	Кф	Кк1	Кк2
Технические критерии оценки ресурсоэффективности							
1. Способствует увеличению скорости выполнения задач	0,14	5	4	4	0,75	0,51	0,41
2. Простота внедрения	0,13	5	5	5	0,65	0,65	0,65
3. Простота эксплуатации	0,11	4	4	4	0,4	0,4	0,4
4. Надежность	0,9	4	3	4	0,4	0,3	0,3
5. Результативность	0,09	5	4	4	0,45	0,36	0,36

Экономические критерии оценки эффективности							
1. Конкурентоспособность продукта	0,09	4	4	4	0,36	0,36	0,36
2. Цена	0,10	3	3	4	0,5	0,3	0,4
3. Предполагаемый срок эксплуатации	0,10	5	4	5	0,42	0,45	0,45
4. Послепродажное обслуживание	0,07	5	3	3	0,33	0,21	0,21
5. Срок выхода на рынок	0,08	4	5	5	0,12	0,4	0,60
Итого	1	44	39	42	4,38	3,94	4,14

Таким образом, конкурентоспособность данной разработки составила 4,38, в то время как у конкурентных методов этот показатель составил 3,94 и 4,14 соответственно. Результаты показывают, что данная разработка является конкурентоспособной и имеет преимущества по некоторым критериям.

5.2.3 SWOT-анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – представляет собой комплексный анализ научно-исследовательского проекта. SWOT-анализ применяют для исследования внешней и внутренней среды проекта.

Он проводится в несколько этапов.

Первый этап заключается в описании сильных и слабых сторон проекта, в выявлении возможностей и угроз для реализации проекта, которые проявились или могут появиться в его внешней среде.

Второй этап состоит в выявлении соответствия сильных и слабых сторон научно-исследовательского проекта внешним условиям окружающей среды. Это соответствие или несоответствие должны помочь выявить степень необходимости проведения стратегических изменений. В рамках второго этапа были составлены таблицы 5.2-5.5.

В рамках третьего этапа составляется итоговая матрица SWOT-анализа, представленная в таблице 5.6.

Таблица 5.2 - Интерактивная матрица сильных сторон и возможностей

Сильные стороны проекта				
Возможности		C1	C2	C3

проекта	B1	+	+	+
	B2	+	-	+
	B3	+	+	-

Таблица 5.3 - Интерактивная матрица слабых сторон и возможностей

Слабые стороны проекта					
Возможности проекта		Сл1	Сл2	Сл3	Сл4
	B1	+	+	-	+
	B2	+	+	-	-
	B3	+	+	+	-

Таблица 5.4 - Интерактивная матрица сильных сторон и угроз

Сильные стороны проекта				
Угрозы проекта		C1	C2	C3
	У1	-	+	+
	У2	-	+	-

Таблица 5.5 - Интерактивная матрица слабых сторон и угроз

Слабые стороны проекта					
Угрозы проекта		Сл1	Сл2	Сл3	Сл4
	У1	-	-	+	-
	У2	+	+	-	+

Таблица 5.6 - Матрица SWOT

	<p>Сильные стороны научно-исследовательского проекта:</p> <p>C1. Низкая стоимость;</p> <p>C2. Ординарность изготовления;</p> <p>C3. Наличие оборудования, позволяющего проведения экспериментов;</p> <p>C4. Конкурентоспособность продукта.</p>	<p>Слабые стороны научно-исследовательского проекта:</p> <p>Сл1. Трудоемкость процесса разработки и тестирования продукта.</p> <p>Сл2. Низкий уровень прибыльности;</p> <p>Сл3. Разнообразие систем распознавания номерных знаков</p>
<p>Возможности:</p> <p>B1. Повышения</p>	<p>Удешевленная разработка за счет использования низкой</p>	<p>Воздействие на объем спроса путем предоставления</p>

<p>конкурентоспособности потенциальных потребителей разработанного проекта.</p> <p>В2. Один алгоритм, который подходит для решения сложных краевых задач для обыкновенных дифференциальных уравнений второго порядка.</p>	<p>стоимости оборудования и возможности динамично пополнять базы автомобильных номеров транспортных средств не только для личных целей, но и также для доступа транспортных средств специальных назначений, что может привлечь спрос не только частных лиц, но и специальных служб (таких как пожарная охрана, медицинских учреждений).</p>	<p>широкого спектра услуг.</p> <p>Установление связей с потребителями для продвижения создаваемого продукта.</p>
<p>Угрозы:</p> <p>У1. Отсутствие спроса на конечный продукт.</p> <p>У2. Появление проектов-конкурентов, разработанных на основе метода конечных разностей</p> <p>У3. Наличие альтернативных подходов.</p>	<p>Значительной угрозой является появление продуктов заменителей, так как производительность и качество техники быстро растет, а также рост цен на материалы, так как цена на данное оборудование зависит от курса доллара.</p> <p>Необходимо предусмотреть планы закупок.</p>	<p>Данная разработка может быть не востребована вовсе в силу ненужности. В данном случае необходим анализ причин развития конкурентов и перенятия их опыта</p>

На основе анализа, представленного в таблице выше (Таблица 5.6 – SWOT-анализ) необходимо особое внимание уделить количеству закупаемого оборудования, так как курс рубля по отношению к доллару является крайне неустойчивым. В связи с низкой стоимости предоставляемой услуги, ожидается повышенный спрос, способствующий заинтересовать государственные структуры предоставлением возможности беспрепятственного доступа транспортных средств специального назначения на территорию предприятий. Данные возможности могут снизить слабые

стороны проекта за счет повышенного спроса и государственной поддержки. Угроза со стороны конкурентов являются самыми непредсказуемыми и критичными, на которые всегда необходимо обращать особое внимание. Но так как данная разработка является собственной, то данные угрозы можно снизить за счет низкой стоимости, а также уделять большое внимание на внедрение данной технологии установление связей с потенциальными покупателями, иначе данная разработка будет невостребованной.

5.2.4 Диаграмма Исикавы

Основной проблемой в задачах распознавания автомобильных номеров является неточное распознавание символов номерных знаков. Для описания данной проблемы была использована диаграмма Исикавы.

Диаграмма Исикавы – графический способ исследования и определения наиболее существенных причинно-следственных взаимосвязей между факторами и последствиями в исследуемой ситуации или проблеме.

Диаграмма представлена на рисунке 5.1.

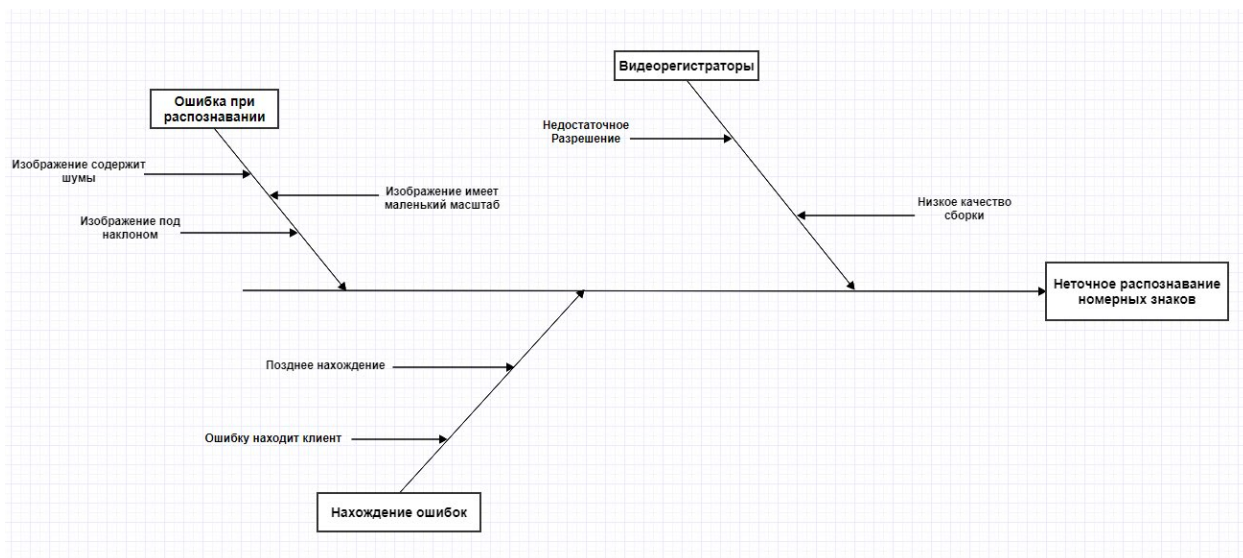


Рисунок 5.1 – Диаграмма Исикавы

5.2.5 Определение возможных альтернатив проведения исследования

Для того, чтобы определить возможные альтернативные варианты для данного исследования, на основе составленной морфологической матрицы (Таблица 5.7 –Морфологическая матрица) определены морфологические характеристики объекта.

Таблица 5.7 – Морфологическая матрица

	1	2
Платформа	Raspberry Pi 3 model B	ASUS Tinker Board
А. Модель процессора	BCM2837	Rockchip RK3288
Б. Количество ядер процессора	4	4
В. Частота процессора	1200 МГц	1800 МГц
Г. Тип оперативной памяти	LPDDR3	LPDDR3
Д. Объем оперативной памяти	1 Гб	2 Гб
Е. Модель видеочипа	Videocore 4	Mali T764
Ж. Интерфейсы периферии	USB 2.0 x4, GPIO, micro USB, micro SD	USB 2.0 x4, GPIO, micro USB, micro SD

Возможные варианты решения поставленной проблемы с позиции функционального содержания и ресурсосбережения:

1. A1B1B1Г1Д1E1Ж1 – RaspberryPi 3 modelB; BCM2837; Количество ядер процессора 4; 1200 МГц; LPDDR3; Объем оперативной памяти 1 Гб; Videocore 4; USB 2.0 x4, GPIO, microUSB, microSD.

2. A2B2B2Г2Д2E2Ж2 – ASUSTinkerBoard; RockchipRK3288; Количество ядер процессора 4; 1800 МГц; LPDDR3; Объем оперативной памяти 2 Гб; MaliT764; USB 2.0 x4, GPIO, microUSB, microSD.

5.3 Оценка готовности проекта к коммерциализации

Далее необходимо оценить степень готовности научного – проекта к коммерциализации и определить уровень собственных знаний для проведения (завершения). В таблице 5.8 представлена специальная форма, содержащая показатели о степени проработанности проекта с позиции коммерциализации и компетенциям разработчика проекта.

Таблица 5.8 - Оценка степени готовности научного проекта к коммерциализации

№ п/п	Наименование	Степень проработанности научного проекта	Уровень имеющихся знаний у разработчика
1.	Определен имеющийся научно-технический задел	4	4
2.	Определены перспективные направления коммерциализации научно-технического задела	5	4
3.	Определены отрасли и технологии (товары, услуги) для предложения на рынке	4	4
4.	Определена товарная форма научно-технического задела для представления на рынок	4	2
5.	Определены авторы и осуществлена охрана их прав	4	2
6.	Проведена оценка стоимости интеллектуальной собственности	4	3
7.	Проведены маркетинговые исследования рынков сбыта	4	4
8.	Разработан бизнес-план коммерциализации научной разработки	4	3
9.	Определены пути продвижения научной разработки на рынок	4	2
10.	Разработана стратегия (форма) реализации научной разработки	3	2

11.	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	1	1
12.	Проработаны вопросы использования услуг инфраструктуры поддержки, получения льгот	4	3
13.	Проработаны вопросы финансирования коммерциализации научной разработки	3	2
14.	Имеется команда для коммерциализации научной разработки	3	3
15.	Проработан механизм реализации научного проекта	4	3
	ИТОГО БАЛЛОВ	55	42

При проведении анализа по таблице 5.8, производится оценка по двум направлениям: степень проработанности научного – проекта и уровень имеющихся знаний у разработчика, по пятибалльной шкале.

Оценка готовности научного проекта к коммерциализации (или уровень имеющихся знаний у разработчика) определяется по формуле:

$$B_{\text{сум}} = \sum B_i, \quad (2)$$

где $B_{\text{сум}}$ – суммарное количество баллов по каждому направлению;

B_i – балл по i -му показателю.

Значение $B_{\text{сум}}$ позволяет определить готовность научной разработки и ее разработчика к коммерциализации. Степень $B_{\text{сум}}$ в данной научной разработке считается перспективной (средняя), следовательно, необходимо увеличивать объемы инвестирования и улучшить направление проведение оценки стоимости интеллектуальной стоимости, повысить уровень компетенций недостающих разработчику в данном вопросе и предусмотреть возможности привлечения требуемых специалистов в команду проекта.

5.3.1 Методы коммерциализации результатов научно-технического исследования

При коммерциализации научно-технических разработок продавец (а это, как правило, владелец соответствующих объектов интеллектуальной собственности), преследует вполне определенную цель, которая во многом

зависит от того, куда в последующем он намерен направить (использовать, вложить) полученный коммерческий эффект. Это может быть получение средств для продолжения своих научных исследований и разработок (получение финансирования, оборудования, уникальных материалов, других научно-технических разработок и т. д.), одноразовое получение финансовых ресурсов для каких-либо целей или для накопления, обеспечение постоянного притока финансовых средств, а также их различные сочетания. При этом время продвижения товара на рынок во многом зависит от правильности выбора метода коммерциализации. Для разработанного проекта инжиниринг является самым оптимальным методом коммерциализации. Инжиниринг – самостоятельный вид коммерческих операций предполагает предоставление на основе договора инжиниринга одной стороной, именуемой консультантом, другой стороне, именуемой заказчиком, комплекса или отдельных видов инженерно-технических услуг, связанных с проектированием, строительством и вводом объекта в эксплуатацию, с разработкой новых технологических процессов на предприятии заказчика, усовершенствованием имеющихся производственных процессов вплоть до внедрения изделия в производство и даже сбыта продукции.

5.4 Инициализация проекта

На этом этапе принято выполнить определенные работы для успешного запуска проекта. Здесь следует определить цели и результаты проекта, определить рабочую группу, определить ограничения и допущения. Вся эта информация заносится в документ, называемый уставом проекта.

Уставом проекта называется документ, который определяет бизнес-потребности и текущее понимание потребностей заказчика, а также новый продукт, услугу или результат, который планируется создать.

5.4.1 Цели и результат проекта

Инициация проекта определяет изначальные цели проекта или содержание работы. Также, в данном разделе определяются изначальные экономические – финансовые затраты. По данной работе также надо определить внутренние и внешние заинтересованные стороны проекта, которые влияют на общий результат научного проекта.

Таблица 5.9 – Заинтересованные стороны проекта

Заинтересованные стороны проекта	Ожидания заинтересованных сторон
НИ ТПУ	Проведение исследований по данной теме с целью использования настоящей разработки в образовательных целях, а также использование данной разработки в качестве основы под иные проекты, выполняемые на базе ОЭФ.
В сфере безопасности дорожного движения	Использование данной разработки, которая позволит повысить контроль и безопасности дорожного движения, и используется в различных областях: охраняемые зоны, предприятия, контроль дорожного движения, заправочные станции, автомобильные стоянки, контроль въезда и выезда и т.п.

Таблица 5.10 – Цели и результат проекта

Цели проекта:	Разработка и программная реализация алгоритма распознавания номерных знаков транспортных средств на полученном локализованном изображении.
Ожидаемые результаты проекта:	Разработанный алгоритм для распознавания номерных пластин
Критерии приемки результата проекта:	Автоматизировать процесс нахождения номерного знака, и учета автомобилей при въезде и выезде с охраняемых зон.

Требования к результату проекта:	Требование:
	Интуитивно понятный упрощенный интерфейс, высокая обработка и распознавания номерных пластин.

5.4.2 Организационная структура проекта

На данном этапе работы необходимо решить следующие вопросы: кто будет входить в рабочую группу данного проекта, определить роль каждого участника в данном проекте, а также прописать функции, выполняемые каждым из участников и их трудозатраты в проекте.

Таблица 5.11 – Рабочая группа проекта

№ п/п	ФИО, основное место работы, должность	Роль в проекте	Функции	Трудо-затраты, час.
1	Доцент ОЭФ ИЯТШ НИТПУ Семёнов М.Е.	Руководитель проекта	<ul style="list-style-type: none"> – составление и утверждение научного задания; – календарное планирование работ; – оценка эффективности полученных результатов. 	25
2	Магистр ОЭФ ИЯТШ НИТПУ Кенджаев Г.О.	Исполнитель проекта	исполнение всех работ по проекту	558

В ходе реализации научного проекта, помимо магистранта задействован руководитель проекта:

Руководитель проекта – отвечает за реализацию проекта в пределах заданных ограничений по ресурсам, координирует деятельность участников

проекта. В рамках данного проекта эту роль выполняет руководитель магистерской диссертации.

Исполнитель по проекту – специалист, выполняющий отдельные работы по проекту. В рамках этого проекта исполнителем является магистрант.

5.5 План работ по научно-исследовательскому проекту

5.5.1 Структура работ в рамках научного исследования

В рамках планирования научного проекта построен календарный график проекта, представленный в таблице 5.12.

Приняты следующие сокращения:

НР – научный руководитель, Семёнов М.Е.

И – исполнитель, Кенджаев Г.О.

В рамках планирования научного проекта был построен календарный план проекта в виде линейного графика в таблице (таблица 5.12) и его иллюстрация работы в виде диаграммы Ганта (таблица 5.13). Диаграммы Ганта строилась с разбивкой по месяцам и декадам (10 дневки) который охватывает весь период времени выполнения научного проекта.

Таблица 5.12 - Календарный план проекта

Код	Название	Длительность, дни	Дата начала работ	Дата окончания работ	Состав участников
1	Постановка целей и задач	1	1.02	1.02	НР
2	Анализ существующих программ	5	2.02	8.02	М
3	Исследование и выбор метода	4	9.02	14.02	М, НР

	сегментации				
4	Исследование и выбор метода распознавания	4	15.02	20.02	М, НР
5	Исследование и выбор метода обработки изображения	4	22.02	27.02	М, НР
6	Выбор программного обеспечения, библиотеки, языка программирования	6	28.02	7.03	М, НР
7	Сбор обучающей и тестовой выборки	8	8.03	17.03	М
8	Реализация метода сегментации	19	20.03	14.04	М
9	Реализация метода распознавания	15	17.04	8.05	М
10	Улучшения алгоритма	5	11.05	17.05	М
11	Тестирование	3	18.05	22.05	М, НР
12	Оформление расчетно-пояснительной записки	4	13.05	28.05	М
13	Оформление графического материала	2	28.05	29.05	М
14	Подведение итогов	2	30.05	31.05	М, НР
И т о г о:		82			
Инженер:		81	1.02	31.05	
Научный руководитель:		24			

Таблица 5.13 – Диаграмма Ганта

Код	Вид работ	Исполнители	Т _к , кол, дн.	Продолжительность выполнения работ, месяц														
				02			03			04			05					
				1	2	3	1	2	3	1	2	3	1	2	3			
1	Постановка целей и	НР	1															

	задач																		
2	Анализ существующих программ	М	5	■															
3	Исследование и выбор метода сегментации	М, НР	4	■	■														
4	Исследование и выбор метода распознавания	М, НР	4		■	■													
5	Исследование и выбор метода обработки изображения	М, НР	4		■	■													
6	Выбор программного обеспечения, библиотеки, языка программирования	М, НР	6			■	■												
7	Сбор обучающей и тестовой выборки	М	8			■	■	■											
8	Реализация метода сегментации	М	19					■	■	■									
9	Реализация метода распознавания	М	15							■	■	■							
10	Улучшения алгоритма	М	5														■		
11	Тестирование	М, НР	3															■	■
12	Оформление расчетно-пояснительной записки	М	4																■
13	Оформление графического материала	М	2																■
14	Подведение итогов	М, НР	2																■

Научный руководитель (НР) ■

Магистрант (М) ■

4.6 Бюджет научного исследования

Общий бюджет научно исследовательской работы состоит из 4 видов планируемых расходов: материальные расходы, вознаграждения, отчисления на социальные нужды и накладные расходы.

5.5.2 Расчет затрат на специальное оборудование для научных (экспериментальных) работ

В данной статье приведены все затраты, связанные с приобретением специального оборудования необходимого для проведения работ по конкретной теме. Определение стоимости спецоборудования производится по действующим прейскурантам, а в ряде случаев по договорной цене.

Все расчеты по приобретению спецоборудования и оборудования, имеющегося в организации, но используемого для каждого исполнения конкретной темы приводятся в таблице ниже (Таблица 5.16 – Материальные затраты на приобретение спецоборудования для научных работ)

Таблица 5.16 – Материальные затраты и затраты на спецоборудования для научных работ

№ п/п	Наименование	Количество	Цена за ед., руб.	Затраты на материалы, руб.
1	Ноутбук DNS 0154207	1	35990	35990
2	Microsoft Windows 7 Professional RU x32/x64	1	8290	8290
3	Пакет Microsoft Office 2019 Home and Student RU x32/x64	1	4600	4600
4	Пакет Microsoft Visual Studio Professional 2019 Single OLV	1	27837	27837
5	Канцелярские принадлежности	1	822	822

Итого	77539
-------	-------

Расходы на приобретение спецоборудование для научных работ составляют 77539 рублей.

5.5.3 Основная заработная плата

Статья включает основную заработную плату работников, непосредственно занятых выполнением проекта, (включая премии, доплаты) и дополнительную заработную плату.

$$C_{зп} = Z_{осн} + Z_{доп}, \quad (3)$$

где $Z_{осн}$ – основная заработная плата;

$Z_{доп}$ – дополнительная заработная плата.

Основная заработная плата ($Z_{осн}$) руководителя от университета:

$$Z_{осн} = Z_{дн} \cdot T_{раб}, \quad (4)$$

где $Z_{осн}$ – основная заработная плата одного работника; T_p – продолжительность работ, выполняемых научно-техническим работником, раб. дн.; $Z_{дн}$ – среднедневная заработная плата работника, руб.

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{дн} = \frac{Z_m \cdot M}{F_d}, \quad (5)$$

где Z_m – месячный должностной оклад работника, руб.;

M – количество месяцев работы без отпуска в течение года:

при отпуске в 24 раб. дня $M = 11,2$ месяца, 5-дневная неделя;

при отпуске в 48 раб. дней $M = 10,4$ месяца, 6-дневная неделя;

F_d – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн.

Таблица 5.17 – Баланс времени

Показатели рабочего времени	Руководитель	Магистрант
Календарное число	366	366

дней		
Количество нерабочих дней (выходные дни и праздничные дни)	51 и 14	51 и 14
Потери рабочего времени (отпуск, больничные)	52	52
Действительный годовой фонд рабочего времени	249	249

Месячный должностной оклад работника:

$$Z_m = Z_b \cdot (k_{пр} + k_d) \cdot k_p, \quad (6)$$

где Z_b – базовый оклад, руб.;

$k_{пр}$ – премиальный коэффициент, (определяется Положением об оплате труда);

k_d – коэффициент доплат и надбавок (в НИИ и на промышленных предприятиях – за расширение сфер обслуживания, за профессиональное мастерство, за вредные условия: определяется Положением об оплате труда);

k_p – районный коэффициент, равный 1,3 (для Томска).

Месячный должностной оклад работника:

$$Z_m = Z_b \cdot k_p \quad (7)$$

где Z_b – базовый оклад, руб.; k_p – районный коэффициент, равный 1,3 (для Томска).

Таблица 5.18 – Расчёт основной заработной платы

Исполнители	Z_b , руб.	$k_{пр}$	k_d	k_p	Z_m , руб.	$Z_{дн}$, руб.	T_p , раб. дн.	$Z_{осн}$, руб.
Руководитель	33664	0,3	0,2	1,3	65644,8	2741,8	5,5	15079,9
Магистрант	12663	-	-	1,3	16463,2	682,14	90	61392,6

5.5.4 Дополнительная заработная плата научно-производственного персонала

Дополнительная заработная плата включает оплату за непроработанное время (очередной и учебный отпуск, выполнение государственных обязанностей, выплата вознаграждений за выслугу лет и т.п.) и рассчитывается исходя из 10-15% от основной заработной платы, работников, непосредственно участвующих в выполнении темы:

$$Z_{\text{доп}} = k_{\text{доп}} \cdot Z_{\text{осн}}, \quad (8)$$

где $Z_{\text{доп}}$ – дополнительная заработная плата, руб.;

$k_{\text{доп}}$ – коэффициент дополнительной зарплаты;

$Z_{\text{осн}}$ – основная заработная плата, руб.

В таблице 5.19 приведена форма расчёта основной и дополнительной заработной платы.

Таблица 5.19 – Расчёт дополнительной заработной платы

Заработная плата	Руководитель	Магистрант
Основная заработная плата	15079,9	61392,6
Дополнительная заработная плата	1809,6	7367,1
Итого по статье $C_{\text{зп}}$	9176,7	

Общая сумма дополнительной заработной платы исполнителей проекта составила 9176,7 рубля.

5.5.5 Отчисления на социальные нужды

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников.

Величина отчислений во внебюджетные фонды определяется исходя из следующей формулы:

$$C_{\text{внеб}} = k_{\text{внеб}} \cdot (З_{\text{осн}} + З_{\text{доп}}), \quad (9)$$

где $k_{\text{внеб}}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

Таблица 5.20 – Отчисления во внебюджетные фонды

Исполнители	Основная ЗП, руб	Доп. ЗП, руб	Сумма отчисления, руб
Руководитель	15079,9	1809,6	4577,1
Магистрант	61392,6	7367,1	18633,9
		Итого	23211

5.5.6 Прочие прямые затраты

В данном пункте рассчитываются затраты на электричество за время работы над проектом.

Затраты на электроэнергию рассчитываются по формуле:

$$C_{\text{эл.об}} = P_{\text{об}} \cdot t_{\text{об}} \cdot Ц_{\text{э}}, \quad (10)$$

где $P_{\text{об}}$ – мощность, потребляемая оборудованием, кВт; $t_{\text{об}}$ – время работы оборудования, час; $Ц_{\text{э}}$ – тариф на 1 кВт·час. Для ТПУ $Ц_{\text{э}} = 5.8$ руб./кВт · час.

Время работы оборудования вычисляется на основе итоговых данных таблицы 5.19 для инженера из расчета, что продолжительность рабочего дня равна 8 часов.

$$t_{\text{об}} = T_{\text{рД}} \cdot K_t \quad (11)$$

где K_t – коэффициент использования оборудования по времени, $K_t = 0,9$.

Мощность, потребляемая оборудованием, определяется по формуле:

$$P_{\text{Т}} = P_{\text{ном}} \cdot K_{\text{С}}$$

где K_C – коэффициент загрузки. Для технологического оборудования малой мощности $K_C = 1$.

Итоги расчет затрат на электроэнергию приведены на таблице 5.21.

Таблица 5.21 – Затраты на электроэнергию технологическую

Наименование оборудования	Время работы Оборудования $t_{об}$, час	Потребляемая мощность $P_{об}$, кВт	Затраты $C_{эл.об}$, руб.
Персональный компьютер инженера	162	0,09	2000

5.5.7 Накладные расходы

В эту статью включаются затраты на управление и хозяйственное обслуживание, которые могут быть отнесены непосредственно на конкретную тему. Кроме того, сюда относятся расходы по содержанию, эксплуатации и ремонту оборудования, производственного инструмента и инвентаря, зданий, сооружений и др. В расчетах эти расходы принимаются в размере 70 - 90 % от суммы основной заработной платы научно-производственного персонала данной научно-технической организации.

Накладные расходы составляют 80-100 % от суммы основной и дополнительной заработной платы, работников, непосредственно участвующих в выполнение темы.

Расчет накладных расходов ведется по следующей формуле:

$$C_{накл} = k_{накл} \cdot (Z_{осн} + Z_{доп}), \quad (13)$$

где $k_{накл}$ – коэффициент накладных расходов.

$$C_{накл} = 0,3 \cdot (15079,9 + 1809,6 + 61392,6 + 7367,1) = 25694,76 \text{ руб.}$$

5.5.8 Формирование бюджета затрат НИИ

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в

качестве нижнего предела затрат на разработку научно-технической продукции.

На основании полученных данных по отдельным статьям затрат составляется калькуляция плановой себестоимости проекта. Определение бюджета затрат на научно-исследовательский проект приведен в таблице ниже:

Таблица 5.22 – Калькуляция затрат по статьям

Наименование статьи	Сумма, руб
1. Специальное оборудование для научных работ	77539
2. Затраты на основную ЗП	76472,5
3. Затраты на дополнительную ЗП	9 176,7
4. Отчисления на социальные нужды	23 211
5. Затраты на электроэнергию	2000
6. Накладные расходы	25 694,76
7. Прочие прямые затраты	84,56
Итого	214178,52

5.6 Реестр рисков проекта

Идентифицированные риски проекта включают в себя возможные неопределенные события, которые могут возникнуть в проекте и вызвать последствия, которые повлекут за собой нежелательные эффекты.

Таблица 5.23 – Реестр рисков

№	Риск	Потенциальное воздействие	Вероятность наступления (1-5)	Влияние риска (1-5)	Уровень риска*	Способы смягчения риска	Условия наступления
1	Отсутствие спроса со стороны ключевых	Отсутствие возможности продаж продукции	2	4	Средний	- Разработка политик	Отсутствие заинтересованности в конкретном

	потребителе й для приобретени я продукта					и по продвиж ению продукц ии на рынке - подготов ка к смене ориентац ии продукта	продукте
2	Технологиче ские риски, связанные с использован ием ПО, сторонних разработчико в	Незаверше нность продукта	1	2	Низкий	Рассмотр ение аналогич ных, ПО использу емых в разработ ке	Изменение политики распростране ния используемо го ПО
3	Сбои в разработке готовой продукции	Проблемы с внедрение алгоритма	2	2	Низкий	Проводи ть полное модульн ое тестиров ание	Отсутствие полноценног о тестирование
4	Конкуренция на рынке	Снижение спроса на разработан ное алгоритм	2	4	Средний	Подгото вка к снижени ю стоимост и продукц ии. Поиск дополни тельных клиентов из других категори й	Высокая цена продукта. Отсутствие необходимог о функционала

Уровень риска может быть: высокий, средний или низкий в зависимости от вероятности наступления и степени влияния риска.

5.7 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

5.7.1 Оценка абсолютной эффективности исследования

В основе проектного подхода к инвестиционной деятельности предприятия лежит принцип денежных потоков (cashflow). Особенностью является его прогнозный и долгосрочный характер, поэтому в применяемом подходе к анализу учитываются фактор времени и фактор риска. Для оценки общей экономической эффективности используются следующие основные показатели:

- чистая текущая стоимость (NPV);
- индекс доходности (PI);
- внутренняя ставка доходности (IRR);
- срок окупаемости (DPP).

Чистая текущая стоимость (NPV) – это показатель экономической эффективности инвестиционного проекта, который рассчитывается путём дисконтирования (приведения к текущей стоимости, т.е. на момент инвестирования) ожидаемых денежных потоков (как доходов, так и расходов).

Расчёт NPV осуществляется по следующей формуле:

$$NPV = \sum_{t=1}^n \frac{ЧДП_{опt}}{(1+i)^t} - I_0$$

где: $ЧДП_{опt}$ – чистые денежные поступления от операционной деятельности;

I_0 – разовые инвестиции, осуществляемые в нулевом году;

t – номер шага расчета ($t= 0, 1, 2 \dots n$)

n – горизонт расчета;

i – ставка дисконтирования (желаемый уровень доходности инвестируемых средств).

Расчёт NPV позволяет судить о целесообразности инвестирования денежных средств. Если $NPV > 0$, то проект оказывается эффективным.

Расчет чистой текущей стоимости представлен в таблице 5.24. При расчете рентабельность проекта составляла 20%, амортизационные отчисления 10 %. Бюджет проекта = 138901,5

Таблица 5.24. Расчет чистой текущей стоимости по проекту в целом

№	Наименование показателей	Шаг расчета				
		0	1	2	3	4
1	Выручка от реализации, руб.	0	257014.224	257014.224	257014.224	257014.224
2	Итого приток, руб.	0	257014.224	257014.224	257014.224	257014.224
3	Инвестиционные издержки, руб.	- 214178.52	0	0	0	0
4	Операционные затраты, руб.	0	0	0	0	0
5	Налогооблагаемая прибыль	0	257014.224	257014.224	257014.224	257014.224
6	Налоги 20 %, руб.	0	51402,84	51402,84	51402,84	51402,84
7	Итого отток, руб.	- 214178.52	51402,84	51402,84	51402,84	51402,84
8	Чистая прибыль, руб.	0	205611.4	205611.4	205611.4	205611.4
9	Чистый денежный поток (ЧДП), руб.	- 214178.52	227029.3	227029.3	227029.3	227029.3
10	Коэффициент дисконтирования (КД)	1	0,833333	0,694444	0,578704	0,482253
11	Чистый	-	189191	157659	131382.76	109485.56

	дисконтированный доход (ЧДД), руб.	214178.52				
12	Σ ЧДД	587718.32				
12	Итого NPV, руб.	373539,8				

Коэффициент дисконтирования рассчитан по формуле:

$$КД = \frac{1}{(1 + i)^t}$$

где: i – ставка дисконтирования, 20 %;

Таким образом, чистая текущая стоимость по проекту в целом составляет 373539,8 рублей, что позволяет судить о его эффективности

Индекс доходности (PI) – показатель эффективности инвестиции, представляющий собой отношение дисконтированных доходов к размеру инвестиционного капитала. Данный показатель позволяет определить инвестиционную эффективность вложений в данный проект. Индекс доходности рассчитывается по формуле:

$$PI = \sum_{t=1}^n \frac{ЧДП_t}{(1 + i)^t} / I_0$$

где: ЧДП - чистый денежный поток, млн. руб.;

I_0 – начальный инвестиционный капитал, млн. руб.

$$PI = \frac{587718,32}{214178,52} = 2,744$$

Таким образом $PI > 1$, то проект является эффективным.

Значение ставки, при которой обращается в нуль, носит название «внутренней ставки доходности» или IRR. Формальное определение «внутренней ставки доходности» заключается в том, что это та ставка дисконтирования, при которой суммы дисконтированных притоков денежных средств равны сумме дисконтированных оттоков или =0. По разности между IRR и ставкой дисконтирования i можно судить о запасе

экономической прочности инвестиционного проекта. Чем ближе IRR к ставке дисконтирования i , тем больше риск от инвестирования в данный проект.

$$\sum_{t=1}^n \frac{\text{ЧДП}_{\text{оп}t}}{(1 + IRR)^t} = \sum_{t=0}^n \frac{I_t}{(1 + IRR)^t}$$

Между чистой текущей стоимостью (NPV) и ставкой дисконтирования (i) существует обратная зависимость. Эта зависимость представлена в таблице 5.25 и на рисунке 5.2.

Таблица 5.25. Зависимость NPV от ставки дисконтирования

№	Наименование показателя	0	1	2	3	4	NPV, руб
1	Чистые денежные потоки, руб	- 214178,5 2	227029,3	227029,3	227029,3	227029,3	
2	Коэффициент дисконтирования						
	0,2	1	0,833	0,694	0,579	0,482	
	0,3	1	0,769	0,592	0,455	0,350	
	0,4	1	0,714	0,510	0,364	0,260	
	0,5	1	0,667	0,444	0,296	0,198	
	0,6	1	0,625	0,391	0,244	0,153	
	0,7	1	0,588	0,346	0,204	0,120	
	0,8	1	0,556	0,309	0,171	0,095	
	0,9	1	0,526	0,277	0,146	0,077	
	1	1	0,500	0,250	0,125	0,063	
	1,1	1	0,476	0,227	0,108	0,051	
	1,2	1	0,455	0,207	0,094	0,043	
	1,3	1	0,435	0,189	0,082	0,036	
	1,4	1	0,417	0,174	0,072	0,030	
3	Дисконтированный денежный доход, руб.						
	0,2	- 214178,52	189191,08	157659,24	131382,70	109485,58	159361,56
	0,3	- 214178,52	174637,92	134336,86	103336,05	79489,27	63443,06

	0,4	- 214178,52	162163,79	115831,28	82736,63	59097,59	-8527,76
	0,5	- 214178,52	151352,87	100901,91	67267,94	44845,29	-63989,03
	0,6	- 214178,52	141893,31	88683,32	55427,08	34641,92	-107711,41
	0,7	- 214178,52	133546,65	78556,85	46209,91	27182,30	-142861,33
	0,8	- 214178,52	126127,39	70070,77	38928,21	21626,78	-171603,89
	0,9	- 214178,52	119489,11	62889,00	33099,48	17420,78	-195458,68
	1	- 214178,52	113514,65	56757,33	28378,66	14189,33	-215517,07
	1,1	- 214178,52	108109,19	51480,57	24514,56	11673,60	-232579,13
	1,2	- 214178,52	103195,14	46906,88	21321,31	9691,50	-247242,21
	1,3	- 214178,52	98708,39	42916,69	18659,43	8112,80	-259959,73
	1,4	- 214178,52	94595,54	39414,81	16422,84	6842,85	-271081,00

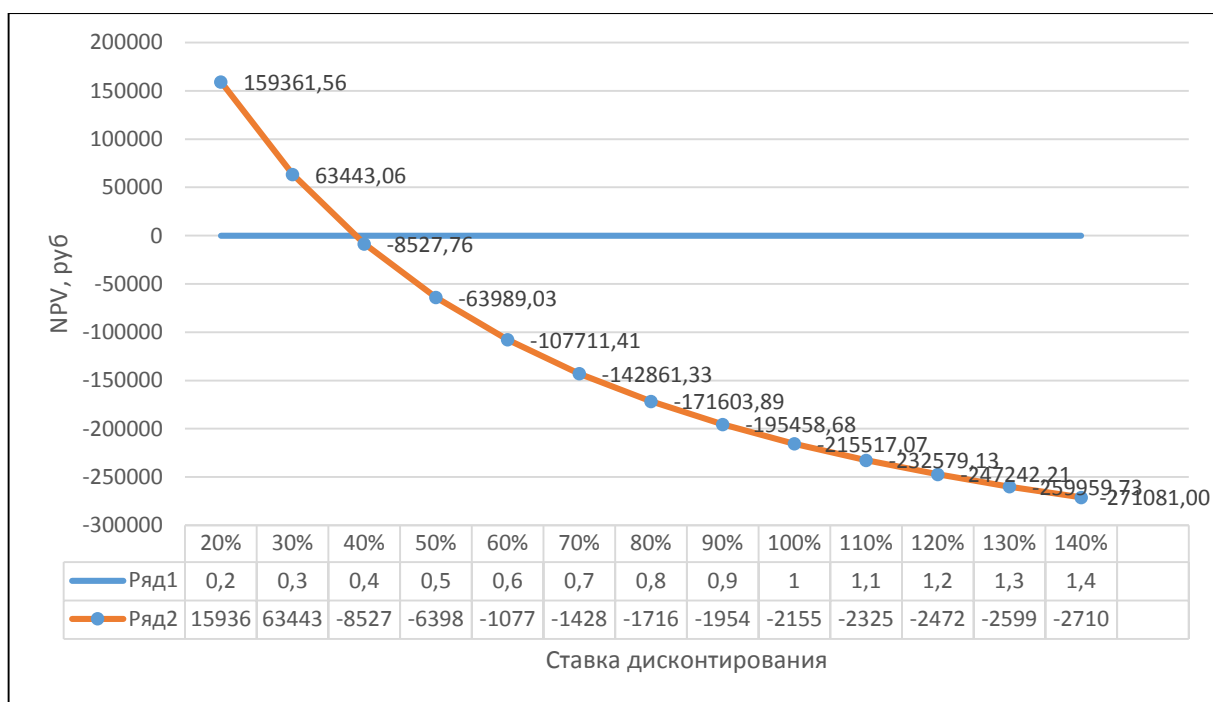


Рисунок 5.2 – Зависимость NPV от ставки дисконтирования

Из таблицы и графика следует, что по мере роста ставки дисконтирования чистая текущая стоимость уменьшается, становясь отрицательной. Значение ставки, при которой NPV обращается в нуль, носит название «внутренней ставки доходности» или «внутренней нормы прибыли». Из графика получаем, что IRR составляет 0,38664.

Запас экономической прочности проекта: $38,664\% - 20\% = 18,66\%$

Как отмечалось ранее, одним из недостатков показателя простого срока окупаемости является игнорирование в процессе его расчета разной ценности денег во времени.

Этот недостаток устраняется путем определения дисконтированного срока окупаемости. То есть это время, за которое денежные средства должны совершить оборот.

Наиболее приемлемым методом установления дисконтированного срока окупаемости является расчет кумулятивного (нарастающим итогом) денежного потока (таблица 5.26).

Таблица 5.26. Дисконтированный срок окупаемости

№	Наименование показателя	Шаг расчета				
		0	1	2	3	4

1	Дисконтированный денежный доход ($i=0,20$), руб.	- 214178.5 2	189191	157659	131382.76	109485.56
2	То же нарастающим итогом, руб.	- 214178.5 2	-24987,52	132671,4 8	264054,24	373539,8
3	Дисконтированный срок окупаемости	$PP_{диск} = 1 + (24987,52 / 157659) = 1,16$ года				

Социальная эффективность научного проекта (таблица 5.27) учитывает социально-экономические последствия осуществления научного проекта для общества в целом или отдельных категорий населения или групп лиц, в том числе как непосредственные результаты проекта, так и «внешние» результаты в смежных секторах экономики: социальные, экологические и иные внеэкономические эффекты.

Таблица 5.27. Критерий социальной эффективности

ДО	ПОСЛЕ
Трата большого количества времени для выполнения рутинных задач, что ведет к избыточной трате электроэнергии	Экономия электроэнергии за счет использования чат-бота для рутинных задач, так как тратиться меньше времени за компьютером для решения тех же задач.

5.8 Оценка сравнительной эффективности исследования

Определение эффективности происходит путем расчета интегрального показателя эффективности научного исследования. Он рассчитывается на основе двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный показатель финансовой эффективности научного исследования получают в ходе оценки бюджета затрат трех (или более) вариантов исполнения научного исследования. Для этого наибольший интегральный показатель реализации технической задачи принимается за

базу расчета (как знаменатель), с которым соотносятся финансовые значения по всем вариантам исполнения.

Интегральный финансовый показатель разработки определяется как:

$$I_{\Phi}^p = \frac{\Phi_{pi}}{\Phi_{max}}, \quad (14)$$

где I_{Φ}^p – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i – го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского

проекта (в т.ч. аналоги). За максимально возможную стоимость исполнения следует принять 280000 рублей.

Полученная величина интегрального финансового показателя разработки показывает численное увеличение/уменьшение бюджета затрат разработки в разгах. Интегральный показатель ресурсоэффективности вариантов исполнения объекта определяется следующим образом:

$$I_T^p = \sum a_i \cdot b_i \quad (15)$$

где I_T^p – интегральный показатель ресурсоэффективности вариантов;

a_i – весовой коэффициент i – го варианта исполнения разработки;

b_i – балльная оценка i – го варианта исполнения разработки;

устанавливается экспертным путем по выбранной шкале оценивания;

Расчет интегрального показателя ресурсоэффективности представлен ниже:

Таблица 5.28– Расчет интегрального показателя ресурсоэффективности

Критерии	Весовой коэффициент параметра	Исп.1: магистрант	Исп.2: конкурент
Способствует росту производительности труда пользователя	0,3	4	3
Возможность применения любым предприятием	0,15	4	4
Требует наличия	0,2	5	5

исторических данных			
Простота применения	0,1	4	5
Конкурентоспособность (с другими системами)	0,25	5	4
ИТОГО	1	4,45	3,9

$$I_{\text{риск.1}} = 4 * 0,3 + 4 * 0,15 + 5 * 0,2 + 4 * 0,1 + 5 * 0,25 = 4,45;$$

$$I_{\text{риск.2}} = 3 * 0,3 + 4 * 0,15 + 5 * 0,2 + 5 * 0,1 + 4 * 0,25 = 3,9;$$

$$I_{\text{max}} = 4,45.$$

Интегральный показатель эффективности вариантов исполнения разработки определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{\text{финр}}^p = \frac{I_{\text{т}}^p}{I_{\text{ф}}^p} \quad (16)$$

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Сравнительная эффективность проекта определяется по формуле 17:

$$\mathcal{E}_{\text{ср}} = \frac{I_{\text{финр}}^p}{I_{\text{max}}} \quad (17)$$

Сравнительная эффективность разработки представлена в таблице 5.29:

Таблица 5.29 – Расчет интегрального показателя ресурсоэффективности

№ п/п	Показатели	Проект магистранта	Проект конкурента
1	Интегральный финансовый показатель разработки	0,81	1,08
2	Интегральный показатель ресурсоэффективности разработки	4,45	3,9
3	Интегральный показатель эффективности	5,49	3,61
4	Сравнительная эффективность вариантов исполнения	1,52	0,66

Таким образом, сравнивая значения интегральных показателей эффективности можно сделать следующие выводы:

1. Научный проект требует больше ресурсов для реализации;
2. Научный проект более эффективен чем конкурент.

Выводы по главе "Финансовый менеджмент, ресурсоэффективность и ресурсосбережение"

В ходе выполнения части работы по финансовому менеджменту, ресурсоэффективности и ресурсосбережению был проведен анализ разрабатываемого исследования.

1. Было проведено планирование НИИ, а именно: определена структура и календарный план работы, трудоемкость, составлена ленточная диаграмма Ганта, и определен бюджет научно-исследовательской работы. Результаты соответствуют требованиям ВКР по срокам и иным параметрам.
2. Оценен коммерческий потенциал и перспективность проведения исследования. Полученные результаты говорят о потенциале и перспективности на уровне выше среднего.
3. Оценены риски проекта. Главной проблемой может стать снижение спроса на разработанное алгоритм. Эту проблему можно предотвратить созданием оптимального алгоритма, который подходит ко всем прикладным задачам.
4. Определена эффективность исследования в разрезах ресурсной, финансовой, бюджетной и экономической эффективности.

ЗАКЛЮЧЕНИЕ

В данной работе проведен литературный обзор задачи распознавания автомобильных номеров, проведен обзор существующих систем распознавания автомобильных номеров с оценкой качества, после чего сформулирована цель работы и определен перечень задач, подлежащих разработке.

Предложенный алгоритм распознавания автомобильных номеров реализован на языке программирования C# с использованием библиотеки компьютерного зрения EmguCV.

В результате работы был разработан и реализован программное обеспечение с графическим интерфейсом пользователя. Программное обеспечение имеет приемлемую точность распознавания и высокое быстродействие. На текстовых выборка процент точности распознавания составил 89 процента, средняя скорость обработки от 500 мс до 730 мс в зависимости от расстояния до объекта.

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

Список публикаций студента

1. Кенджаев, Г.О. Распознавание автомобильных номерных знаков на локализованном изображении / Г.О. Кенджаев // Точная наука. – 2020. - №80. (принято к печати)

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ Р 50577-93. Знаки государственные регистрационные транспортных средств. Типы и основные размеры. Технические требования. – М.: ИПК Издательство стандартов, 2002. – 27 с.
2. Автомаршал [Электронный ресурс] [URL:http://avtomarshal.ru/](http://avtomarshal.ru/). свободный. – Заглавие с экрана. – Язык рус. Дата обращения: 8.06.2020.
3. SL-Traffic. Безопасный город [Электронный ресурс] [URL:http://goal.ru/](http://goal.ru/). свободный. – Заглавие с экрана. – Язык рус. Дата обращения: 4.06.2020.
4. SecurOSAuto [Электронный ресурс] [URL:http://www.iss.ru/products/intelligent/auto/](http://www.iss.ru/products/intelligent/auto/). (дата обращения 22.05.2020 г.)
5. Новые Технологии :: Компьютерные телевизионные системы безопасности CVS (Computer Video Security) [Электронный ресурс] URL: <http://www.cvsnt.ru/>, свободный. – Заглавие с экрана. – Язык рус. Дата обращения: 10.06.2020.
6. Быстрое размытие по Гауссу // Хабрахабр [Электронный ресурс] [URL:https://habr.com/ru/post/151157/](https://habr.com/ru/post/151157/) свободный. (дата обращения 17.05.2020 г.)
7. Бинаризация изображений // Recog.ru [Электронный ресурс] URL: <http://recog.ru/blog/applied/15.html>. (дата обращения 26.05.2020 г.)
8. Детектор границ Канны // Хабрахабр [Электронный ресурс] [URL:https://habr.com/ru/post/114589/](https://habr.com/ru/post/114589/). (дата обращения 26.05.2020 г.)
9. Сегментация изображений // Хабрахабр [Электронный ресурс] URL: <https://habr.com/ru/post/128768/>. (дата обращения 26.05.2020 г.)
10. Работа каскада Хаара в картинках: теория и практика.//Хабрахабр [Электронный ресурс] [URL:https://habr.com/ru/company/recognitor/blog/228195/](https://habr.com/ru/company/recognitor/blog/228195/). (дата обращения 29.05.2020 г.)
11. Метод Виолы-Джонса как основа для распознавания лиц // Хабрахабр [Электронный ресурс] URL: <https://habr.com/ru/post/133826/> (дата обращения 23.05.2020 г.)
12. Обучение OpenCV каскада Хаара // Хабрахабр [Электронный ресурс] URL: <https://habr.com/ru/post/208092/>. (дата обращения 25.05.2020 г.)
13. A Real-Time Mobile Vehicle License Plate Detection and Recognition Kuо-Ming Hung and Ching-Tang Hsieh :

- гистограммный подход при распознавании номеров [Электронный ресурс]
URL: <http://www2.tku.edu.tw/~tkjse/13-4/09-IE9722.pdf/> (дата обращения 20.05.2020 г.)
14. Гистограмма яркости изображения. [Электронный ресурс]
URL: https://studbooks.net/2250867/informatika/gistogramma_yarkosti_izobrazeniya/. (дата обращения 18.05.2020 г.)
 15. Хайкин С. Нейронные сети : полный курс : пер. с англ. / С. Хайкин. – 2-е изд., испр.. – М. [и др.]: Вильямс, 2006. – 1103 с.
 16. Комарцова Л. Нейрокомпьютеры : учебное пособие для вузов / Л. Г. Комарцова, А. В. Максимов. – 2-е изд., перераб. и доп.. – М.: Изд-во МГТУ, 2004. – 399 с.
 17. Классификатор изображений на основе сверточной сети [Электронный ресурс] URL: <http://mechanoid.kiev.ua/ml-lenet.html> (дата обращения 27.05.2020 г.)
 18. OpenCV // Wikipedia [Электронный ресурс]
URL: <https://ru.wikipedia.org/wiki/OpenCV/>. (дата обращения 27.05.2020 г.)
 19. EmguCV [Электронный ресурс]
URL: http://www.emgu.com/wiki/index.php/Main_Page/ (дата обращения 27.05.2020 г.)
 20. Фильтр Гаусса // Wikipedia [Электронный ресурс]
URL: https://ru.wikipedia.org/wiki/Фильтр_Гаусса. (дата обращения 24.05.2020 г.)
 21. Интегральное представление изображений // Хабрахабр [Электронный ресурс] URL: <https://studfiles.net/preview/6234768/page:3/>. (дата обращения 25.05.2020 г.)
 22. Гонсалес Р., Вудс Р. Цифровая обработка изображений. Издание 3-е, исправленное и дополненное. – М.: Техносфера, 2012. David M. E., Shervin E., Khvedchenia I. Mastering OpenCV with Practical Computer Vision Projects by // Bag-gio Published by Packt Publishing, 2012.
 23. Свириин И., Ханин А. Некоторые аспекты автоматического распознавания автомобильных номеров // Алгоритмы безопасности, 2010, № 3. С. 26-29.
 24. N. Otsu. A threshold selection method from gray-level histograms // IEEE Trans. Sys., Man., Cyber. : journal. — 1979. — Vol. 9. — P. 62—66.
 25. Ping-Sung Liao and Tse-Sheng Chen and Pau-Choo Chung. A Fast Algorithm for Multilevel Thresholding // J. Inf. Sci. Eng.. — 2001. — Т. 17. — С. 713—727.
 26. Huang, Deng-Yuan (2009). "Optimal multi-level thresholding using a two-stage Otsu optimization approach". *Pattern Recognition Letters*. **30** (3): 275–284.
 27. Liao, Ping-Sung (2001). "[A fast algorithm for multilevel thresholding](#)" (PDF). *J. Inf. Sci. Eng.* **17** (5): 713–727.

28. Апальков И.В., Хрящев В.В. Удаление шума из изображений на основе нелинейных алгоритмов с использованием ранговой статистики. – Ярославский государственный университет, 2007.
29. Оператор Собеля [Электронный ресурс] // Википедия: свободная энцикл. – Электрон. дан. – [Б. м.], 2012. URL: <https://ru.wikipedia.org/wiki/> (дата обращения 18.05.2020 г.)
30. Вежнев В., Барина О. Алгоритмы сегментации изображений: автоматическая сегментация [Электронный ресурс]. – URL: <http://www.cgm.computergraphics.ru.postman.ru/content/view/147>. (дата обращения 28.05.2020 г.)
31. Nagpal, A. Review based on data clustering algorithms / A. Nagpal, A. Jatani, D. Gaur // IEEE Conference, Information & Communication Technologies (ICT). – IEEE, 2013. – P. 298-303.
32. Fahad, A. A survey of clustering algorithms for big data: Taxonomy and empirical analysis / A. Fahad, N. Alshatri, Z. Tari et al. // IEEE Transactions, Emerging Topics in Computing. – 2014. – Vol. 2. – No. 3. – P. 267-279.
33. Фисенко В.Т., Фисенко Т.Ю., Компьютерная обработка и распознавание изображений: учеб. пособие. - СПб: СПбГУ ИТМО, 2008. – 192 с.
34. Canny J.F. Finding edges and lines in images. Master's thesis. MIT, Cambridge, USA, 1983, pp.50–67.
35. Метод AvgSdv из библиотеки компьютерного зрения Emgu [Электронный ресурс] URL: <http://www.emgu.com/wiki/files/2.3.0/document/html/24dc084c-0935-08c9-213f-d677e7dc8e78.htm> свободный. – Заглавие с экрана. – Язык англ. Дата обращения: 28.05.2020
36. СанПиН 2.2.2/2.4.1340-03. Санитарно-эпидемиологические правила и нормативы «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».
37. Закон Российской Федерации “О защите населения и территорий от чрезвычайных ситуаций природного и техногенного характера”
38. СП 52.13330.2016 Естественное и искусственное освещение.
39. ГОСТ 12.1.003-2014 ССБТ. Шум. Общие требования безопасности.
40. СанПин 2.2.4.548-96 Гигиенические требования к микроклимату производственных помещений.
41. ГОСТ 12.0.003-2015 ССБТ. Опасные и вредные производственные факторы. Классификация.
42. СанПиН 2.2.1/2.1.1.1200-03. Санитарно-защитные зоны и санитарная классификация предприятий, сооружений и других объектов
43. ГОСТ Р 51768-2001. Ресурсосбережение. Обращение с отходами. Методика определения ртути в ртутьсодержащих отходах. Общие требования.

44. ГОСТ 12.1.019-2017 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты.

Приложение А
(справочное)

Development of an algorithm and recognition program for a car registration license
plate in a localized area

Студент

Группа	ФИО	Подпись	Дата
0BM82	Кенджаев Голибджон Орифович		

Руководитель ВКР

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

Консультант-лингвист отделения иностранных языков ШБИП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Маркова Наталия Александровна			

There is a large number of industrial license plate recognition software. This section provides a brief analytical review of some software products. The description of each product contains information about the name, scope and recognition technologies used.

Introduction to the subject area

The current state standard of the Russian Federation defines 22 types of state registration marks (GTP) installed on vehicles [1]. The standard defines the main dimensions, as well as technical requirements for the RSA. The technical requirements described in the standard are aimed at ensuring road safety, life, health, safety of property of the population and environmental protection.

Depending on the registration plates used, vehicles are divided into the following groups:

1 - vehicles owned by legal entities and citizens of the Russian Federation, legal entities and citizens of foreign states, except those assigned to the group no. 3, as well as stateless persons;

2 - vehicles of military units and formations under the jurisdiction of the Russian Federation and formed in accordance with applicable laws;

3 - vehicles owned by diplomatic missions, consular posts, international (interstate) organizations;

4 - vehicles temporarily allowed to participate in traffic;

5 - vehicles owned by the internal affairs bodies of the Russian Federation.

In this paper, the recognition of the first type of GTP with a three-digit and two-digit region code according to the current RF standard [1] is considered. The standard establishes the following types of registration marks (Figure 1.1):

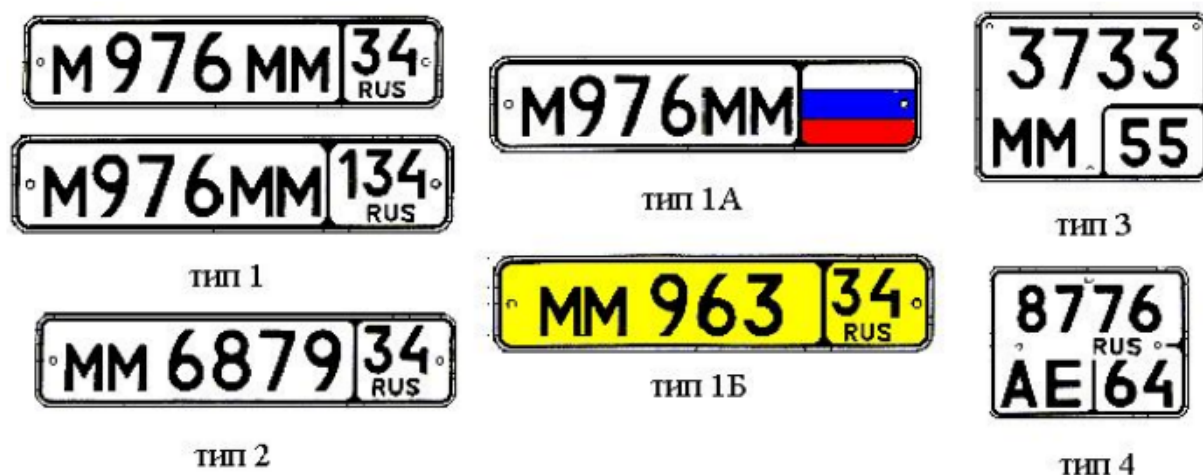


Figure 1.1. Types of registration marks of the 1st group

Recognition of the remaining types of gas distribution devices is not considered in this paper. Examples of type 1 GTP with a two-digit and three-digit registration region code are shown in Figures 1.2 and 1.3.



Figure 1.2. Type mark «1» with a two-digit region code [1]



Figure 1.3. Type mark «1» with a three-digit region code [1]

The main part of license plates in Russia are standard marks of the 1993 model, the definition of which contains State Standard R 50577-93 [1].

Car license plates of diplomatic vehicles, Ministry of Internal Affairs vehicles, trailers, military vehicles have a slight difference from the standard.

The standard license plates of the Russian Federation must contain: three letters, three digits, region number two or three digits, the flag of Russia and the inscription RUS. Letters mean a series of license plate, and numbers mean a number. For use on signs, 12 letters of the Cyrillic alphabet are allowed, which are in the Latin alphabet - A, B, E, K, M, H, O, P, C, T, Y and X.

For each region of the Russian Federation, there is a common number for all vehicles registered in this region.

Characteristics of license plate recognition systems

Among the license plate recognition software, one can single out: "Marshal", "Special Lab-Traffic", "SecurOSAuto". Table 1 shows the characteristics of these programs.

Table 1. License plate recognition systems

Operating system	Windows 7SP1/8.1/10 (32/64 bits) Windows Server 2008R2/2012/2012R2/2016	Windows Server Linux Server	Windows Windows Server
Price, RUB	29 900	30 000	48 110
Recognition probability, %	98	99	98
Application area	automating the operation of parking lots, checkpoints and checkpoints, metering vehicles on highways,	security, control of entry / exit of vehicles from	list of tasks of law enforcement agencies,

	weights, car washes and gas stations, as well as for controlling the access of cars to the territory of apartment buildings and suburban settlements	parking lots, collective garages, territories with limited access	including the search for vehicles, automation of commercial and municipal parking lots, control of toll roads, weight control, traffic control at sensitive facilities, border and customs control
--	--	---	--

Algorithms and image processing methods

For image processing, there are many methods and algorithms; here is a brief description of the methods and algorithms used in the work.

A Gaussian filter is an electronic filter in which the Gaussian function is a pulse transition function. The Gaussian filter is designed so that the time constant is maximum and there is no overshoot in the transition function. This behavior is closely related to the fact that the Gaussian filter has the minimum possible group delay. A Gaussian filter is usually used in digital form for processing two-dimensional signals (images) in order to reduce noise [20].

The next algorithm is the **Haar detector**. Its main advantage is speed. Fast image processing makes it easy to process video sequences. The Haar detector is

used to detect most classes of objects. These include faces and other parts of the body of people, license plates, passers-by, road symbols, animals, etc.

The classifier is formed on Haar primitives by the method of calculating characteristic values. For training, a set of “correct” images with a pre-selected area in the image is supplied to enter the classifier, after which primitives are searched and the attribute value is calculated [12]. Mostly, rectangular primitives are used (Figure 1.4).

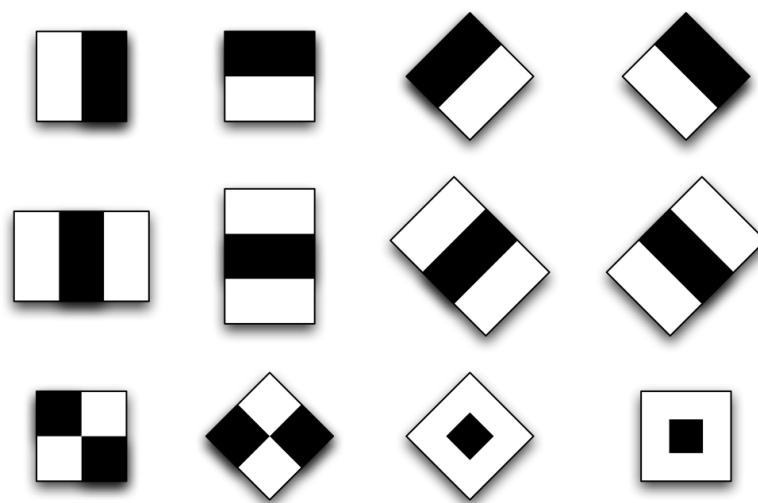


Figure 1.4. Haar Primitives

In the review, we draw attention to the histogram method. To construct histograms of image brightness, the binarization operation is used: we select black and white pixels. Subsequently, the binarization is calculated in each column, the number of black pixels and on the basis of the information obtained is built histogram image. The principle of histogram construction is shown in Figure 1.5.

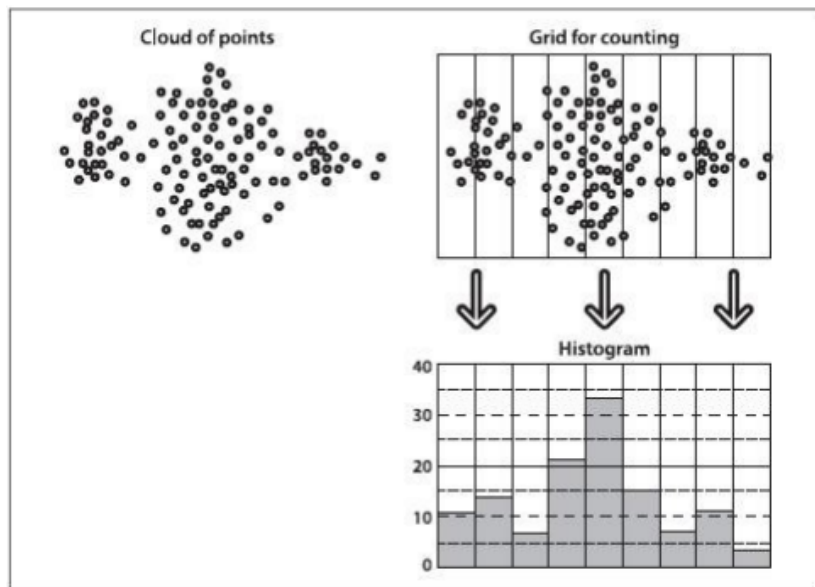


Figure 1.5. The principle of constructing a brightness diagram

An example of a binarized license plate and its histogram (orange line) are shown in Figures 1.6 and 1.7.



Figure 1.6. Binarized car number



Figure 1.7. A histogram of the brightness of the license plate

Figure 1.7 shows that there are sharp fluctuations in the curve, which is used to determine the boundaries (segmentation) of the license plate.

Methods of segmentation

Image segmentation is the process of dividing a digital image into a large number of areas (various objects) that have a similar texture or color. The result of segmentation is a set of areas (contours) extracted from the image. All pixels from the same area are similar in some respects: color, texture, or intensity. Neighboring areas differ from each other by the characteristics listed above. Different approaches to finding the boundaries between regions are based on the heterogeneity of the values of brightness intensity. Thus, the choice of image segmentation method depends on the difficulty that needs to be solved.

Segmentation algorithms are classified as follows:

1. Threshold algorithms;
2. Algorithms sprawl regions;
3. Boundary algorithms;
4. Segmentation based on clustering.

Threshold algorithms set some threshold restrictions. After conversion in the image matrix, image elements having a degree of brightness greater than the threshold will have a value of 1, less than the threshold value of 0. The best threshold value is formed based on the analysis of histograms. Next, multi-threshold segmentation is applied [24].

Threshold algorithms make it possible to segment ordinary images, but do not give a good result in images with the presence of uneven lighting, flashes and various noise. There are algorithms to eliminate the effects of these defects, which analyze the weighted values of the extremes, or rather analyze the intensity and gradient [25].

For the correct application of the algorithm, it is necessary to avoid “deviations” when choosing a threshold value.

1. It is necessary to control the identity of the distribution in the bright and dark areas of the histogram of brightness tightly.

2. It is necessary to break the image into small components so that the brightness histogram has a vivid expression of the extrema.

3. Elements should be quite large. The size of the statistical sample should make it possible to satisfactorily estimate the positions of the extrema and describe the neighborhood.

The Otsu method [24] is used to calculate the binarization threshold for a grayscale image. In the usual form, the algorithm returns one intensity threshold, which delimits pixels into two classes: useful and background. Calculating a similar threshold so that the intraclass dispersion of intensity is minimal. The Otsu method is a one-dimensional discrete analogue of Fisher's discriminant analysis, is associated with the Jenks optimization method and is equivalent to the globally optimal k-average performed on the intensity histogram. Expansion to a multi-level threshold value is described in the article [25], the development of ideas is described in [26, 27].

Otsu Algorithm

Let the monochrome image be given:

$$G(i, j), i = \overline{1, Height}, j = \overline{1, Width}.$$

1. Set the repetition counter $k = 0$.
2. Calculate the histogram $p(l)$ of the image and the frequency $N(l)$ for each level of image intensity G .
3. Calculate the initial values for $\omega_1(0), \omega_2(0), \mu_1(0), \mu_2(0)$.
4. For each value $t = \overline{1, \max(G)}$ halftone is the horizontal axis of the histogram:
 - 4.1. Update $\omega_1, \omega_2, \mu_1, \mu_2$.
 - 4.2. We compute $\sigma_b^2(t) = \omega_1(t), \omega_2(t)[\mu_1(t), \mu_2(t)]^2$.

- 4.3. If $\sigma_b^2(t)$ is greater than the existing one, then we remember σ_b^2 and the value of the threshold t .
- 4.4. The desired threshold corresponds to the maximum $\sigma_b^2(t)$

$$N_T = \sum_{i=0}^{\max(G)} p(i), \quad (3)$$

$$\omega_1(t) = \frac{\sum_{i=0}^{t-1} p(i)}{N_T} = \sum_{i=0}^{t-1} N(i), \quad \omega_2(t) = 1 - \omega_1(t). \quad (4)$$

$$\mu_T = \frac{\sum_{i=0}^{\max(G)} i \cdot p(i)}{N_T} = \sum_{i=0}^{\max(G)} i \cdot N(i), \quad (5)$$

$$\mu_1(t) = \frac{\sum_{i=0}^{t-1} i \cdot p(i)}{N_T \cdot \omega_1(t)} = \frac{\sum_{i=0}^{t-1} i \cdot N(i)}{\omega_1(t)}, \quad \mu_2(t) = \frac{\mu_T - \mu_1(t) \cdot \omega_1(t)}{\omega_2(t)} \quad (6)$$



Figure 1.8. An example of binarization with an Otsu threshold

Growth algorithms for regions are effectively applied on images that have areas with stable connectivity within individual segments. As a result of the operation of such algorithms, a grouping of neighboring elements with similar or similar brightness levels occurs, which as a result are connected into homogeneous regions [28].

The disadvantages of the algorithms for increasing regions are primarily attributed to the fact that they distinguish common fragments, often without showing information about changes in brightness within regions and probable boundaries. However, these algorithms have proven to be more effective when working with noisy images.

Boundary algorithms are algorithms consisting in finding points lying on the boundaries of regions [30]. Pixel brightness values are basic for determining boundaries. Commonly used features include texture and gradient.

The boundaries of the objects in the image significantly reduce the amount of data that needs to be processed, and at the same time saves important information about the objects in the image, their shape, size, quantity.

Segmentation algorithms have specific applications that depend on:

- input data
- recognition requirements
- calculation volume
- speed.

When choosing a segmentation algorithm, the most popular and well-known methods were considered. These include cameraman Roberts, Sobel, and Canny's border detector.

Sobel operator - the operator calculates the approximate value of the brightness gradient of the image. As a result of applying the Sobel operator, an image is obtained, each point of which will be either a brightness gradient vector or a norm at this point [29].



Figure 1.9. The result of applying the Sobel operator

As can be seen in Figure 1.9, the operator highlighted the boundaries. It is worth noting that the outline of the license plate and its symbols are clearly visible in the vehicle area.

Robert's operator performs simple and fast calculations of two-dimensional spatial measurement in an image. This method emphasizes areas of high spatial frequency that often correspond to edges. The operator input is a grayscale image. The value of the pixels of the output image at each point implies a certain value of the spatial gradient of the input image at the same point. The detection of boundaries by this method is computationally much simpler than by the Sobel method, but it leads to frequent false triggering of the filter in point bursts of brightness, in fact, which leads to noise in the resulting image.

Canny's edge detector is one of the most popular edge detection algorithms. An important step in this algorithm is the elimination of noise on the circuits, which can significantly affect the result, while it is necessary to keep the boundaries as much as possible. This requires careful selection of the threshold value during processing [34].

Clustering-based segmentation. The advantage of clustering-based segmentation algorithms is the automation of the process of searching for parameter values for class separation. There are many clustering algorithms, among which are Viola-Jones, CURE, BIRCH [31-32], k-means [34], and the support vector method [25]. The indicated algorithms find clusters corresponding to a certain static model.

We also single out an approach for image processing using **convolutional neural networks** [15]. In a convolutional neural network, there are two types of layers alternate: convolutional layers and subsampling layers. The benefits are included.

Приложение В

Листинг программы

Car.cs

```
using System;
using System.Collections.Generic;
using System.Text;
using Emgu.CV;
using Emgu.CV.Structure;

namespace LicensePlateRecognition.Recognize
{
    public class Car
    {
        public Image<Bgr,byte> Original;
        Image<Bgr,byte> Sobel;
        Image<Gray,float> SobelGray;
        public List<Horizontal> Horizont =new List<Horizontal>();
        public List<Plate> Plate =new List<Plate>();
        Image<Gray,byte>[] images =new Image<Gray,byte>[22];
        float[] u;

        List<double[]> Images =new List<double[]>();

        public void Clear()
        {
            Horizont.Clear();
            Plate.Clear();
        }

        public Car()
        {
            images[0]=new Image<Gray,byte>("DigitLetter2\\0.jpg");
            images[1]=new Image<Gray,byte>("DigitLetter2\\1.jpg");
            images[2]=new Image<Gray,byte>("DigitLetter2\\2.jpg");
            images[3]=new Image<Gray,byte>("DigitLetter2\\3.jpg");
            images[4]=new Image<Gray,byte>("DigitLetter2\\4.jpg");
            images[5]=new Image<Gray,byte>("DigitLetter2\\5.jpg");
            images[6]=new Image<Gray,byte>("DigitLetter2\\6.jpg");
            images[7]=new Image<Gray,byte>("DigitLetter2\\7.jpg");
            images[8]=new Image<Gray,byte>("DigitLetter2\\8.jpg");
            images[9]=new Image<Gray,byte>("DigitLetter2\\9.jpg");
            images[10]=new Image<Gray,byte>("DigitLetter2\\A.jpg");
            images[11]=new Image<Gray,byte>("DigitLetter2\\B.jpg");
            images[12]=new Image<Gray,byte>("DigitLetter2\\E.jpg");
            images[13]=new Image<Gray,byte>("DigitLetter2\\K.jpg");
            images[14]=new Image<Gray,byte>("DigitLetter2\\M.jpg");
            images[15]=new Image<Gray,byte>("DigitLetter2\\H.jpg");
            images[16]=new Image<Gray,byte>("DigitLetter2\\O.jpg");
            images[17]=new Image<Gray,byte>("DigitLetter2\\P.jpg");
            images[18]=new Image<Gray,byte>("DigitLetter2\\C.jpg");
            images[19]=new Image<Gray,byte>("DigitLetter2\\T.jpg");
            images[20]=new Image<Gray,byte>("DigitLetter2\\X.jpg");
            images[21]=new Image<Gray,byte>("DigitLetter2\\Y.jpg");
        }

        public Image<Bgr,byte> SetImage(Image<Bgr,byte> image)
        {
```

```

        Original = image;
        SobelGray =image.Convert<Gray,float>();
System.Diagnostics.Stopwatch stop =new System.Diagnostics.Stopwatch();
stop.Start();
        Image<Gray,float> sobelx = SobelGray.Sobel(1,0,3);
        Image<Gray,float> sobely = SobelGray.Sobel(0,1,3);
        SobelGray =(sobelx.Pow(2)+ sobely.Pow(2)).Pow(0.5);
        Sobel = SobelGray.Convert<Bgr,byte>();

Horizontal();
stop.Stop();
return Sobel;
}
privatefloat[] Horiz(Image<Gray,float> image){
float[] u =newfloat[image.Height];
for(int i =0; i <image.Height; i++)
{
float k =0;
for(int j =1; j <image.Width-1; j++)
{
k +=System.Convert.ToSingle(Math.Abs(image.Data[i, j -
1,0]- image.Data[i, j,0]));
}
u[i]= k;
}
return u;
}
publicfloat[] Proizvodnij1(float[] u)
{
float[] u1 =newfloat[u.Length];
u1[0]=(u[0]+ u[1]+ u[2]+ u[3]+ u[4]+ u[5]+ u[6])/7;
u1[1]=(u[0]+ u[1]+ u[2]+ u[3]+ u[4]+ u[5]+ u[6]+ u[7])/8;
u1[2]=(u[0]+ u[1]+ u[2]+ u[3]+ u[4]+ u[5]+ u[6]+ u[7]+u[8])/9;
u1[3]=(u[0]+ u[1]+ u[2]+ u[3]+ u[4]+ u[5]+ u[6]+ u[7]+ u[8]+
u[9])/10;
u1[4]=(u[0]+ u[1]+ u[2]+ u[3]+ u[4]+ u[5]+ u[6]+ u[7]+ u[8]+
u[9]+u[10])/11;
u1[5]=(u[0]+ u[1]+ u[2]+ u[3]+ u[4]+ u[5]+ u[6]+ u[7]+ u[8]+
u[9]+ u[10]+ u[11])/12;
u1[6]=(u[0]+ u[1]+ u[2]+ u[3]+ u[4]+ u[5]+ u[6]+ u[7]+ u[8]+
u[9]+ u[10]+ u[11]+u[12])/13;
for(int i =7; i <u.Length-7; i++)
{
u1[i]=(u[i -6]+ u[i -5]+ u[i -4]+ u[i -3]+ u[i -2]+ u[i -1]+
u[i]+ u[i +1]+
u[i +2]+ u[i +3]+ u[i +4]+ u[i +5]+ u[i +6])/13;
}
u1[u.Length-1]=(u[u.Length -7]+ u[u.Length -6]+ u[u.Length -5]+
u[u.Length -4]+ u[u.Length -3]+ u[u.Length -2]+ u[u.Length -1])/7;
u1[u.Length-2]=(u[u.Length -8]+ u[u.Length -7]+ u[u.Length -6]+
u[u.Length -5]+ u[u.Length -4]+ u[u.Length -3]+ u[u.Length -2]+ u[u.Length -
1])/8;
u1[u.Length-3]=(u[u.Length -9]+ u[u.Length -8]+ u[u.Length -7]+
u[u.Length -6]+ u[u.Length -5]+ u[u.Length -4]+ u[u.Length -3]+ u[u.Length -
2]+ u[u.Length -1])/9;
u1[u.Length-4]=(u[u.Length-10]+ u[u.Length -9]+ u[u.Length -8]+
u[u.Length -7]+ u[u.Length -6]+ u[u.Length -5]+ u[u.Length -4]+ u[u.Length -
3]+ u[u.Length -2]+ u[u.Length -1])/10;
u1[u.Length-5]=(u[u.Length -10]+ u[u.Length -9]+ u[u.Length -8]+
u[u.Length -7]+ u[u.Length -6]+ u[u.Length -5]+ u[u.Length -4]+ u[u.Length -
3]+ u[u.Length -2]+ u[u.Length -1])/11;
u1[u.Length-6]=(u[u.Length -11]+ u[u.Length -10]+ u[u.Length -9]+
u[u.Length -8]+ u[u.Length -7]+ u[u.Length -6]+ u[u.Length -5]+ u[u.Length -
4]+ u[u.Length -3]+ u[u.Length -2]+ u[u.Length -1])/12;

```

```

return u1;
}
public float[] Proizvodnij2(float[] u)
{
float[] u1 = new float[u.Length];
    u1[0] = (u[0] + u[1] + u[2] + u[3] + u[4] + u[5] + u[6]) / 7;
    u1[1] = (u[0] + u[1] + u[2] + u[3] + u[4] + u[5] + u[6] + u[7]) / 8;
    u1[2] = (u[0] + u[1] + u[2] + u[3] + u[4] + u[5] + u[6] + u[7] + u[8]) / 9;
    u1[3] = (u[0] + u[1] + u[2] + u[3] + u[4] + u[5] + u[6] + u[7] + u[8] +
u[9]) / 10;
for (int i = 4; i < u.Length - 4; i++)
{
    u1[i] = (u[i - 3] + u[i - 2] + u[i - 1] + u[i] + u[i + 1] +
u[i + 2] + u[i + 3]) / 7;
}
    u1[u.Length - 1] = (u[u.Length - 7] + u[u.Length - 6] + u[u.Length - 5] +
u[u.Length - 4] + u[u.Length - 3] + u[u.Length - 2] + u[u.Length - 1]) / 7;
    u1[u.Length - 2] = (u[u.Length - 8] + u[u.Length - 7] + u[u.Length - 6] +
u[u.Length - 5] + u[u.Length - 4] + u[u.Length - 3] + u[u.Length - 2] + u[u.Length -
1]) / 8;
    u1[u.Length - 3] = (u[u.Length - 9] + u[u.Length - 8] + u[u.Length - 7] +
u[u.Length - 6] + u[u.Length - 5] + u[u.Length - 4] + u[u.Length - 3] + u[u.Length -
2] + u[u.Length - 1]) / 9;
    u1[u.Length - 4] = (u[u.Length - 10] + u[u.Length - 9] + u[u.Length - 8] +
u[u.Length - 7] + u[u.Length - 6] + u[u.Length - 5] + u[u.Length - 4] + u[u.Length -
3] + u[u.Length - 2] + u[u.Length - 1]) / 10;
return u1;
}
public float MAX(float[] u)
{
float max = u[0];
for (int i = 1; i < u.Length; i++)
{
if (max < u[i])
    max = u[i];
}
return max;
}
public int MAX(int[] u)
{
int max = u[0];
for (int i = 1; i < u.Length; i++)
{
if (max < u[i])
    max = u[i];
}
return max;
}
public void DrawLine(float[] p, float MAX, Image<Bgr, byte> image)
{
for (int i = 0, x = 0; i < p.Length - 1; i++, x++)
{
float j = image.Width * p[i] / MAX;
float j1 = image.Width * p[i + 1] / MAX;
image.Draw(new LineSegment2DF(new System.Drawing.PointF(j, x), new
System.Drawing.PointF(j1, x)), new Bgr(System.Drawing.Color.Orange), 2);
}
}
public float DrawMiddleLineHoriz(float[] p, float MAX, Image<Bgr, byte> image)
{
float sum = 0.0f;
for (int i = 0; i < p.Length; i++)
{

```

```

        sum += p[i];
    }
    sum /= p.Length;
    float h = Sobel.Width * sum / MAX;
    image.Draw(new LineSegment2DF(new System.Drawing.PointF(h,0),new
    System.Drawing.PointF(h, Sobel.Height)),new
    Bgr(System.Drawing.Color.Aqua),2);
    return sum;
}
public float DrawMiddleLineVertical(int index, float Max)
{
    float sum1 = 0.0f;
    for(int j = 0; j < Horizont[index].p.Length; j++)
    {
        sum1 += Horizont[index].p[j];
    }
    sum1 /= Horizont[index].p.Length;
    for(int h1 = 0, x = 0; h1 < Horizont[index].p.Length-1; h1++, x++)
    {
        float j = Horizont[index].sobel.Height * Horizont[index].p[h1]/(Max);
        float j1 = Horizont[index].sobel.Height * Horizont[index].p[h1 + 1]/(Max);
        Horizont[index].sobell.Draw(new LineSegment2DF(new
        System.Drawing.PointF(x, j),new System.Drawing.PointF(x, j1)),new
        Bgr(System.Drawing.Color.Orange),2);
    }
    return sum1;
}
public List<int> SearchMaximums(float[] p, float Sum, float Max, Image<Bgr,byte>
image, bool Vertical)
{
    List<int> maxList = new List<int>();
    bool down = false, up = false;
    for(int i = 0; i < p.Length-1; i++)
    {
        if(p[i]>p[i + 1])
        {
            down = true;
            if(up)
            {
                if(p[i]> Sum)
                    maxList.Add(i);
                up = false;
            }
        }
        if(p[i]<p[i + 1])
        {
            up = true;
            if(down)
            {
                down = false;
            }
        }
    }
    if(!Vertical)
    {
        for(int i = 0; i < maxList.Count; i++)
            image.Draw(new CircleF(new System.Drawing.PointF(Convert.ToSingle(image.Width
            * p[maxList[i]]/ Max), maxList[i]),3),new
            Bgr(System.Drawing.Color.Yellow),6);
    }
    else
    {
        for(int i = 0; i < maxList.Count; i++)

```

```

{
    image.Draw(new CircleF(new
    System.Drawing.PointF(Convert.ToSingle(image.Height * p[maxList[i]]/ Max),
    maxList[i]),3),new Bgr(System.Drawing.Color.Yellow),6);
}
}
return maxList;
}
public List<int> SearchMinimum(float[] p,int index,float Max)
{
    List<int> maxList =new List<int>();
    bool down =false, up =false;
    maxList.Add(0);
    for(int i =0; i <p.Length-1; i++)
    {
        if(p[i]>p[i +1])
        {
            down =true;
        }
        if(up)
        {
            up =false;
        }
    }
    if(p[i]<p[i +1])
    {
        up =true;
    }
    if(down)
    {
        maxList.Add(i);
    }
    down =false;
}
}
}
maxList.Add(p.Length-1);

for(int i =0; i < maxList.Count; i++)
{
    Plate[index].rotate2.Draw(new CircleF(new
    System.Drawing.PointF(maxList[i],
    Convert.ToSingle(Plate[index].rotate2.Height * p[maxList[i]]/ Max)),1),new
    Bgr(System.Drawing.Color.Yellow),2);
}
return maxList;
}
public List<int> SearchMaximum2(float[] p,int index,float Max)
{
    List<int> maxList =new List<int>();
    bool down =false, up =false;

    for(int i =0; i <p.Length-1; i++)
    {
        if(p[i]>p[i +1])
        {
            down =true;
        }
        if(up)
        {
            maxList.Add(i);
            up =false;
        }
    }
    if(p[i]<p[i +1])
    {
        up =true;
    }
}
}
}

```

```

if (down)
{
    down =false;
}
}
}
for(int i =0; i < maxList.Count; i++)
{
    Plate[index].rotate2.Draw(new CircleF(new
System.Drawing.PointF(maxList[i],
Convert.ToSingle(Plate[index].rotate2.Height * p[maxList[i]]/ Max)),1),new
Bgr(System.Drawing.Color.Yellow),2);
}
return maxList;
}
publicvoid Greenze(List<int> maximum,float[] p)
{
for(int i =0; i <maximum.Count; i++)
{
    Horizontal r =new Recognize.Horizontal();
    r.i = maximum[i];
bool endUp =false;
for(int j = maximum[i]+1; j <p.Length-1; j++)
{
if(p[j]<p[j +1])
{
    endUp =true;
r.k2 = j;
break;
}
}
if(!endUp)
{
r.k2 = p.Length -1;
}

bool endDown =false;
for(int j = maximum[i]-1; j >0; j--)
{
if(p[j]< p[j -1])
{
    endDown =true;
r.k1 = j;
break;
}
}
if(!endDown)
{
r.k1 =1;
}
if(Math.Abs(r.k1 - r.k2)>30)
    Horizont.Add(r);
}
}
publicvoidVertical(int index)
{
    Horizont[index].p=newfloat[Horizont[index].sobel.Width];
for(int k =0; k < Horizont[index].sobel.Width; k++)
{
float f =0.0f;
for(int l =1; l < Horizont[index].sobel.Height -1; l++)
{

```

```

        f +=
Convert.ToSingle(Math.Abs(Horizont[index].sobel.Data[l -1, k,0]-
Horizont[index].sobel.Data[l, k,0]));
    }
        Horizont[index].p[k]= f;
    }
}
publicvoidGetPlate(int index,float sum1,float max)
{
bool input =false;
int k1 =0, k2 =0;
for(int w =1; w < Horizont[index].p.Length-1; w++)
{
if(Horizont[index].p[w]> sum1)
{
if(!input)
{
k1 = w;
input =true;
}
}
if(Horizont[index].p[w +1]< sum1 && input)
{
input =false;
k2 = w;

if(k2 - k1 <50)
continue;
//List<int> Maxumums = SearchMaximums(Horizont[index].p, sum1, max,
Horizont[index].sobell, true);
//if (Maxumums.Count < 4)
// continue;

Plate plate =newPlate();
plate.x1 = k1;
plate.x2 = k2;
plate.y1 = Horizont[index].k1;
plate.y2 = Horizont[index].k2;
plate.original= Original.Copy(new System.Drawing.Rectangle(plate.x1,
plate.y1, plate.x2 - plate.x1, plate.y2 - plate.y1));
float sum=0;
for(int hg=plate.y1; hg<plate.y2; hg++)
sum+=u[hg];
double alpha =0.15*(plate.y2 - plate.y1)+0.25/0.5+0.4/ sum
+0.4*Math.Abs((Math.Abs(k1-k2)/Math.Abs(plate.y1 - plate.y2))-5);
if(!Ugol(plate))
Plate.Add(plate);
}
}
}

publicboolUgol(Plate plate)
{
LineSegment2D[] lines =null;
Image<Gray,byte> gray =plate.original.Convert<Gray,byte>();
Image<Gray,float> sobel =gray.Sobel(0,1,3);
CvInvoke.cvConvert(sobel, gray);

try
{
lines =gray.HoughLinesBinary(1, Math.PI /45,50, sobel.Width
/3,0)[0];
}
catch{}
if(lines ==null||lines.Length==0)
{

```

```

return true;
}
double angle = 0;
    LineSegment2D avr = new LineSegment2D();
foreach (LineSegment2D seg in lines)
{
avr.P1 = new System.Drawing.Point(avr.P1.X + seg.P1.X, avr.P1.Y + seg.P1.Y);
avr.P2 = new System.Drawing.Point(avr.P2.X + seg.P2.X, avr.P2.Y + seg.P2.Y);
}
avr.P1 = new System.Drawing.Point(avr.P1.X / lines.Length, avr.P1.Y /
lines.Length);
avr.P2 = new System.Drawing.Point(avr.P2.X / lines.Length, avr.P2.Y /
lines.Length);
    LineSegment2D horizontal = new LineSegment2D(avr.P1, new
System.Drawing.Point(avr.P2.X, avr.P1.Y));
double c = horizontal.P2.X - horizontal.P1.X;
double a = Math.Abs(horizontal.P2.Y - avr.P2.Y);
double b = Math.Sqrt(c * c + a * a);
    angle = (a / b * (180 / Math.PI)) * (horizontal.P2.Y > avr.P2.Y ? 1 :-
1);
plate.Alpha = angle;
plate.Rotate = plate.Original.Rotate(angle, new Bgr(255, 255, 255));
return false;
}
public void Horizontal()
{
System.Diagnostics.Stopwatch stop = new System.Diagnostics.Stopwatch();
stop.Start();
    u = Horiz(SobelGray);
stop.Stop();
    u = Proizvodnij1(u);
    u = Proizvodnij2(u);
float max = MAX(u);
DrawLine(u, max, Sobel);
float sum = DrawMiddleLineHoriz(u, max, Sobel);
    List<int> maximum = SearchMaximums(u, sum, max, Sobel, false);
Greenze(maximum, u);

for (int i = 0; i < Horizont.Count; i++)
{
//Sobel.Draw(new CircleF(new
System.Drawing.PointF(Convert.ToSingle(Sobel.Width * u[Horizont[i].k1] /
max), Horizont[i].k1), 3), new Bgr(System.Drawing.Color.Red), 3);
//Sobel.Draw(new CircleF(new
System.Drawing.PointF(Convert.ToSingle(Sobel.Width * u[Horizont[i].k2] /
max), Horizont[i].k2), 3), new Bgr(System.Drawing.Color.Red), 3);
    Horizont[i].original = Original.Copy(new
System.Drawing.Rectangle(0, Horizont[i].k1, Original.Width, Horizont[i].k2 -
Horizont[i].k1));
    Horizont[i].sobel = Horizont[i].original.Convert<Gray, byte>();
    Horizont[i].sobel1 =
Horizont[i].original.Convert<Bgr, byte>();
    Vertical(i);
    Horizont[i].p = Proizvodnij1(Horizont[i].p);
    Horizont[i].p = Proizvodnij2(Horizont[i].p);
    Horizont[i].p = Proizvodnij2(Horizont[i].p);
float max1 = MAX(Horizont[i].p);
float sum1 = DrawMiddleLineVertical(i, max1);
float h = Horizont[i].sobel1.Height * sum1 / max1;
    Horizont[i].sobel1.Draw(new LineSegment2DF(new
System.Drawing.PointF(0, h), new
System.Drawing.PointF(Horizont[i].sobel1.Width, h)), new
Bgr(System.Drawing.Color.Blue), 2);
}
}

```



```

GetPlate(i, sum1, max1);
}
stop.Stop();
System.Diagnostics.Stopwatch st =new System.Diagnostics.Stopwatch();
st.Start();
    Int32 ret =0;
for(int i =0; i < Plate.Count; i++)
{
    Ugol(Plate[i]);
    Plate[i].rotate2 = Plate[i].rotate.Resize(260,56,
Emgu.CV.CvEnum.INTER.CV_INTER_AREA);
//
    Plate[i].rotatel =Plate[i].rotate2.Convert<Gray,byte>();
    CvInvoke.cvCLAHE(Plate[i].rotatel,5,new
System.Drawing.Size(8,8), Plate[i].rotatel);
//Plate[i].rotatel = Threshold.GrayHCH(Plate[i].rotatel);
    CvInvoke.cvNormalize(Plate[i].rotatel,
Plate[i].rotatel,0,255, Emgu.CV.CvEnum.NORM_TYPE.CV_MINMAX,
Plate[i].rotatel);

    CvInvoke.cvAdaptiveThreshold(Plate[i].rotatel,
Plate[i].rotatel,255,
Emgu.CV.CvEnum.ADAPTIVE_THRESHOLD_TYPE.CV_ADAPTIVE_THRESH_GAUSSIAN_C,
Emgu.CV.CvEnum.THRESH.CV_THRESH_BINARY_INV,15,13);
List<System.Drawing.Rectangle> rec =new List<System.Drawing.Rectangle>();
// Plate[i].rotatel = Plate[i].rotatel.MorphologyEx(new
StructuringElementEx(5, 5, 3, 3,
Emgu.CV.CvEnum.CV_ELEMENT_SHAPE.CV_SHAPE_RECT),
Emgu.CV.CvEnum.CV_MORPH_OP.CV_MOP_CLOSE, 1);
using(Emgu.CV.MemStorage storage =new MemStorage())
{
    int il =0;
    for(Emgu.CV.Contour<System.Drawing.Point> contours =
Plate[i].rotatel.Convert<Gray,byte>().FindContours(
Emgu.CV.CvEnum.CHAIN_APPROX_METHOD.CV_CHAIN_APPROX_SIMPLE,
Emgu.CV.CvEnum.RETR_TYPE.CV_RETR_CCOMP); contours !=null; contours =
contours.HNext)
    {
        Emgu.CV.Contour<System.Drawing.Point> currentContour
= contours;

        if(currentContour.BoundingBox.X
>3&&currentContour.BoundingBox.Width>10&&
currentContour.BoundingBox.Height >10&&
currentContour.BoundingBox.Height <200&&
currentContour.BoundingBox.Width <50&&
currentContour.BoundingBox.Height /
currentContour.BoundingBox.Width <1.5)
        {
            il++;

            Plate[i].rotate2.Draw(currentContour.BoundingBox,new
Bgr(System.Drawing.Color.Aqua),2);
            rec.Add(currentContour.BoundingBox);
        }
    }
    /*if (il < 1)
        {
            Plate.Remove(Plate[i]);
            i--;
            continue;
        }*/
}
}

```

```

for(int io =rec.Count-1; io >=0; io--)
{
for(int ip =0; ip < io; ip++)
{
if(rec[ip].X >rec[ip +1].X)
{
System.Drawing.Rectangle r = rec[ip];
rec[ip]=rec[ip +1];
rec[ip+1]= r;
}
}
}
GetRect(i, rec);
if(!IsPlate(i, rec))
{
Plate.Remove(Plate[i]);
i--;

continue;
}
for(int ih =0; ih <rec.Count; ih++)
{
LetterDigit l =newLetterDigit();
l.LD = Plate[i].rotatel.Copy(rec[ih]);

ret++;

Plate[i].digit.Add(l);
}
Plate[i].px=newfloat[Plate[i].rotatel.Width];
for(int kl =0; kl < Plate[i].rotatel.Width; kl++)
{
for(int kp =0; kp< Plate[i].rotatel.Height; kp++)
{
if(Convert.ToInt32(Plate[i].rotatel[kp, kl].Intensity)==255)
Plate[i].px[kl]++;
}
}
Plate[i].px= Proizvodnij1(Plate[i].px);
Plate[i].px= Proizvodnij2(Plate[i].px);
float maxpx = MAX(Plate[i].px);
for(int hl =0, x =0; hl < Plate[i].px.Length-1; hl++, x++)
{
float j = Plate[i].rotate2.Height* Plate[i].px[hl]/(maxpx);
float j1 = Plate[i].rotate2.Height*Plate[i].px[hl +1]/(maxpx);
Plate[i].rotate2.Draw(new LineSegment2DF(new
System.Drawing.PointF(x, j),new System.Drawing.PointF(x, j1)),new
Bgr(System.Drawing.Color.Orange),2);
}
List<int> min1 = SearchMininum(Plate[i].px, i, maxpx);
/* double sumraz = Plate[i].rotatel.Width / min1.Count - 10;
for (int il = 0; il < min1.Count-1; il++)
{
double ggg= Math.Abs( min1[il+1] - min1[il]);
if (ggg < sumraz)
{
min1.Remove(min1[il + 1]);
il--;
continue;
}
}
*/ if (il % 2 == 0)
Plate[i].rotate2.Draw(new LineSegment2DF(new
System.Drawing.PointF(min1[il], 10), new
System.Drawing.PointF(Convert.ToSingle(min1[il+1]), 10)), new
Bgr(System.Drawing.Color.Aqua), 2);
if (il % 2 == 1)

```

```

        Plate[i].rotate2.Draw(new LineSegment2DF(new
System.Drawing.PointF(min1[il], 15), new
System.Drawing.PointF(Convert.ToSingle(min1[il+1]), 15)), new
Bgr(System.Drawing.Color.Orange), 2);
        }*/
for(int il =0; il < Plate[i].digit.Count; il++)
{
for(int gh =0; gh<images.Length; gh++)
{
        Image<Gray,byte> res =
Plate[i].digit[il].LD.Resize(10,18,
Emgu.CV.CvEnum.INTER.CV_INTER_CUBIC).Sub(images[gh]);

        Plate[i].digit[il]._list.Add(Value2(images[gh],
res));
}
}
//x000xx000
        Plate[i].text =TextLetter(i,0)+ TextDigit(i,1)+
TextDigit(i,2)+ TextDigit(i,3)+ TextLetter(i,4)+ TextLetter(i,5)+
TextDigit(i,6)+ TextDigit(i,7)+ TextDigit(i,8);
        Original.Draw(newSystem.Drawing.Rectangle(Plate[i].x1,
Plate[i].y1, Plate[i].x2 - Plate[i].x1, Plate[i].y2 - Plate[i].y1),new
Bgr(System.Drawing.Color.ForestGreen),2);
}
st.Stop();
}
publicList<System.Drawing.Rectangle> GetRect(int i,
List<System.Drawing.Rectangle> rec)
{
float sum =0.0f;
for(int ij =0; ij <rec.Count; ij++)
{
        sum += rec[ij].Y+ rec[ij].Height;
}
float sredY = sum /rec.Count;
        Plate[i].rotate2.Draw(new LineSegment2DF(new
System.Drawing.PointF(0, sredY),new
System.Drawing.PointF(Plate[i].rotate2.Width, sredY)),new
Bgr(System.Drawing.Color.Green),2);
for(int kl =0; kl <rec.Count; kl++)
{
if(System.Math.Abs(sredY - (rec[kl].Y + rec[kl].Height))>8)
{
rec.Remove(rec[kl]);
}
}
float sredX =0.0f;
for(int kl =0; kl <rec.Count; kl++)
{
        sredX += rec[kl].Width;
}
        sredX = sredX /rec.Count;
for(int lp =0; lp <rec.Count; lp++)
{
if(System.Math.Abs(sredX - rec[lp].Width)>10&& rec[lp].Height / rec[lp].Width
<1.5)
{
rec.Remove(rec[lp]);
}
}
return rec;
}

```

```

bool IsPlate(int i, List<System.Drawing.Rectangle> rect)
{
    float sum = 0.0f;
    for(int ij = 0; ij < rect.Count; ij++)
    {
        sum += rect[ij].Y + rect[ij].Height;
    }
    float sredY = sum / rect.Count;
    int index = 0;
    for(int kl = 0; kl < rect.Count; kl++)
    {
        if(System.Math.Abs(sredY - (rect[kl].Y + rect[kl].Height)) < 8)
            index++;
    }
    float sredX = 0.0f;
    for(int kl = 0; kl < rect.Count; kl++)
    {
        sredX += rect[kl].Width;
    }
    sredX = sredX / rect.Count;
    int index2 = 0;
    for(int lp = 0; lp < rect.Count; lp++)
    {
        if(System.Math.Abs(sredX - rect[lp].Width) < 10)
            index2++;
    }
    if((index == 8 && index2 == 8) || (index == 9 && index2 == 9))
        return true;
    return false;
}

double Value(Image<Gray,byte> im, Image<Gray,byte> sample)
{
    int[] sum = new int[im.Width];
    for(int i = 0; i < im.Width; i++)
    for(int j = 0; j < im.Height; j++)
    {
        if(im[j,i].Intensity > 250)
            sum[i]++;
    }
    int[] sum1 = new int[sample.Width];
    for(int i = 0; i < sample.Width; i++)
    for(int j = 0; j < sample.Height; j++)
    {
        if(sample[j, i].Intensity > 250)
            sum1[i]++;
    }
    double value = 0.0;
    for(int i = 0; i < sum.Length; i++)
    {
        value += System.Math.Abs(sum[i] - sum1[i]);
    }
    return value;
}

double Value1(Image<Gray,byte> im, Image<Gray,byte> im2)
{
    double[] d1 = new double[im.Width];
    for(int i = 0; i < im.Width; i++)
    {
        double sum = 0;
        for(int j = 0; j < im.Height; j++)
        {
            sum += im[j, i].Intensity;
        }
    }
}

```

```

        d1[i]= sum;
    }
    double[] d2 =newdouble[im.Height];
    for(int i =0; i <im.Height; i++)
    {
        double sum =0;
        for(int j =0; j <im.Width; j++)
        {
            sum +=im[i, j].Intensity;
        }
        d2[i]= sum;
    }
    double[] d3 =newdouble[im.Width];
    for(int i =0; i < im2.Width; i++)
    {
        double sum =0;
        for(int j =0; j < im2.Height; j++)
        {
            sum += im2[j, i].Intensity;
        }
        d3[i]= sum;
    }
    double[] d4 =newdouble[im2.Height];
    for(int i =0; i < im2.Height; i++)
    {
        double sum =0;
        for(int j =0; j < im2.Width; j++)
        {
            sum += im2[i, j].Intensity;
        }
        d4[i]= sum;
    }
    double k1 =0;
    for(int i =0; i < d1.Length; i++)
    {
        k1+=System.Math.Abs(d1[i]-d1[3]);
    }
    double k11 =0;
    for(int i =0; i < d2.Length; i++)
    {
        k11 +=System.Math.Abs(d2[i]- d4[3]);
    }

    return1-(k1 + k11)/512000;
}
publicdouble Value2(Image<Gray,byte> im, Image<Gray,byte> sample)
{
    MCvScalar m1, m2;
    Gray gray;
    im.AvgSdv(out gray,out m1);
    Image<Gray,byte> f = im;
    sample.AvgSdv(out gray,out m2);
    return Math.Abs(m1.v0 - m2.v0);
}
/*public double VWV(int digit, Config.Config.Data data)
{
    double add = 0.0;

    // add += System.Math.Abs(conf.data[digit].M1 - data.M1);
    // add += System.Math.Abs(conf.data[digit].M2 - data.M2);
    // add += System.Math.Abs(conf.data[digit].M3 - data.M3);
    // add += System.Math.Abs(conf.data[digit].M4 - data.M4);
    //add += System.Math.Abs(conf.data[digit].M5 - data.M5);
}

```

```

        //add += System.Math.Abs(conf.data[digit].M6 - data.M6);
        // add += System.Math.Abs(conf.data[digit].M7 - data.M7);
        return add;
    }*/
public string TextLetter(int i,int dig)
{
    try
    {
        double min = Plate[i].digit[dig]._list[10];
        int op =10;
        for(int lp =11; lp < Plate[i].digit[dig]._list.Count; lp++)
        {

            double v = Plate[i].digit[dig]._list[lp];
            if(min < v)
            {

                min = v;

            }
            op = lp;
        }
        switch(op)
        {
            case10:return "A";
            case11:return "B";
            case12:return "E";
            case13:return "K";
            case14:return "M";
            case15:return "H";
            case16:return "O";
            case17:return "P";
            case18:return "C";
            case19:return "T";
            case20:return "X";
            case21:return "Y";
            default:return "";
        }
    }
    catch{return "";}
}
public string TextDigit(int i,int dig)
{
    try
    {
        double min = Plate[i].digit[dig]._list[0];
        int op =0;
        for(int lp =1; lp <10; lp++)
        {
            double v = Plate[i].digit[dig]._list[lp];
            if(min < v)
            {

                min = v;
                op = lp;

            }
        }
        switch(op)
        {
            case0:return "0";
            case1:return "1";
            case2:return "2";
            case3:return "3";
            case4:return "4";
            case5:return "5";
            case6:return "6";
        }
    }
}

```

```

case 7: return "7";
case 8: return "8";
case 9: return "9";
default: return "";
}
}
catch{return"";}
}
}
}

```

Threshold.cs

```

using System;
using System.Collections.Generic;
using System.Text;
using Emgu.CV;
using Emgu.CV.Structure;

namespace LicensePlateRecognition.Recognize
{
    public static class Threshold
    {
        #region Метод ОЦУ
        public static Image<Gray,byte>Ocu(Image<Gray,byte> image)
        {
            int min = Convert.ToInt32(image[0,0].Intensity);
            int max = Convert.ToInt32(image[0,0].Intensity);

            int i, temp, temp1;
            int[] hist;
            int histSize;

            int alpha, beta, threshold = 0;
            double sigma, maxSigma = -1;
            double w1, a;

            for(i = 0; i < image.Width; i++)
            {
                for(int j = 0; j < image.Height; j++)
                {
                    temp = Convert.ToInt32(image[j, i].Intensity);
                    if(temp < min) min = temp;
                    if(temp > max) max = temp;
                }

                histSize = max - min + 1;

                hist = new int[histSize];
                for(i = 0; i < histSize; i++)
                {
                    hist[i] = 0;
                }
            }
        }
    }
}

```

```

}

for(i =0; i <image.Width; i++)
{
for(int j =0; j <image.Height; j++)
{
hist[Convert.ToInt32(image[j, i].Intensity)- min]++;
}
}

temp = temp1 =0;
alpha = beta =0;
for(i =0; i <=(max - min); i++)
{
temp += i * hist[i];
temp1 += hist[i];
}

for(i =0; i <(max - min); i++)
{
alpha += i * hist[i];
beta += hist[i];
w1 = Convert.ToDouble(beta) / temp1;
a = alpha / beta - (temp - alpha) / (temp1 - beta);
sigma = w1 * (1- w1) * a * a;
if(sigma > maxSigma)
{
maxSigma = sigma;
threshold = i;
}
}
return image.ThresholdBinary(new Gray(threshold), new Gray(255));
}
#endregion

#region Janovica
public static Image<Gray, byte> Janovica(Image<Gray, byte> image)
{
Image<Gray, byte> im = new Image<Gray, byte>(image.Width,
image.Height, new Gray(255));
for(int i =1; i <image.Width-1; i++)
{
for(int j =1; j <image.Height-1; j++)
{
double Rn = image[j, i -1].Intensity + image[j, i +1].Intensity +
image[j -1, i].Intensity + image[j +1, i].Intensity + 4 * image[j,
i].Intensity;

im[j, i] = new Gray(Convert.ToInt32(image[j, i].Intensity + 0.5 * Rn / 4));
}
}
return im;
}

#endregion

#region Niblek

public static Image<Gray, byte> Niblek(Image<Gray, byte> image)
{
Image<Gray, byte> im = new Image<Gray, byte>(image.Width,
image.Height, new Gray(255));
for(int i =1; i <image.Width-1; i++)

```



```

{
for(int j =1; j <image.Height-1; j++)
{
double d = Math.Pow((Math.Pow(image[j, i].Intensity - image[j, i -
1].Intensity,2)+
Math.Pow(image[j, i].Intensity - image[j, i
+1].Intensity,2)+
Math.Pow(image[j, i].Intensity - image[j -1,
i].Intensity,2)+
Math.Pow(image[j, i].Intensity - image[j +1,
i].Intensity,2))/4,0.5);
im[j, i]=new Gray(image[j, i].Intensity +1* d);
}
}
return im;
}

#endregion

#region SingleScaleRetinex

publicstatic Image<Bgr,byte>SingleScaleRetinex(this Image<Bgr,byte> img,int
gaussianKernelSize,double sigma)
{
var radius = gaussianKernelSize /2;
var kernelSize =2* radius +1;

var ycc =img.Convert<Ycc,byte>();

// Формируем ядро Гауссиана
var sum =0f;
var gaussKernel =newfloat[kernelSize * kernelSize];
for(int i =-radius, k =0; i <= radius; i++, k++)
{
for(int j =-radius; j <= radius; j++)
{
var val =(float)Math.Exp(-(i * i + j * j)/(sigma * sigma));
gaussKernel[k]= val;
sum += val;
}
}
for(int i =0; i < gaussKernel.Length; i++)
gaussKernel[i]/= sum;

// Работаем только с яркостным каналом
var gray =new Image<Gray,byte>(ycc.Size);
CvInvoke.cvSetImageCOI(ycc,1);
CvInvoke.cvCopy(ycc, gray, IntPtr.Zero);

// Размеры изображения
var width =img.Width;
var height =img.Height;

var bmp =gray.Bitmap;
var bitmapData =bmp.LockBits(new
System.Drawing.Rectangle(System.Drawing.Point.Empty, gray.Size),
System.Drawing.Imaging.ImageLockMode.ReadWrite,
System.Drawing.Imaging.PixelFormat.Format8bppIndexed);

unsafe
{
for(var y =0; y < height; y++)
{

```

```

var row =(byte*)bitmapData.Scan0 + y * bitmapData.Stride;
for(var x =0; x < width; x++)
{
var color = row + x;

float val =0;

for(int i =-radius, k =0; i <= radius; i++, k++)
{
var ii = y + i;
if(ii <0) ii =0;if(ii >= height) ii = height -1;

var row2 =(byte*)bitmapData.Scan0 + ii * bitmapData.Stride;
for(int j =-radius; j <= radius; j++)
{
var jj = x + j;
if(jj <0) jj =0;if(jj >= width) jj = width -1;

val +=*(row2 + jj)* gaussKernel[k];

}
}

var newColor =127.5+255* Math.Log(*color / val);
if(newColor >255)
newColor =255;
elseif(newColor <0)
newColor =0;
*color =(byte)newColor;
}
}
}
bmp.UnlockBits(bitmapData);

CvInvoke.cvCopy(gray, ycc, IntPtr.Zero);
CvInvoke.cvSetImageCOI(ycc,0);

returnycc.Convert<Bgr,byte>();
}

#endregion

#region AnisotropicSmoothing

publicstatic Image<Gray,byte>AnisotropicSmoothing(Image<Gray,byte> im,int
smooth)
{
double eps =.1;
Image<Gray,byte> f =new Image<Gray,byte>(im.Size.Width,
im.Size.Height);
bool first =true;
double tmp, min=0, max=0;
for(int y =0; y <f.Size.Height; y++)
{
for(int x=0; x<f.Size.Width; x++){
tmp =im[y, x].Intensity;
if(x >0&& x <im.Size.Width-1&& y >0&& y < im.Size.Height -1)
{
double A = tmp;
double E =im[y +1, x].Intensity;
double S =im[y, x +1].Intensity;
double N =im[y, x -1].Intensity;

```

```

double W =im[y -1, x].Intensity;

double Lw = A - W;
double Le = A - E;
double Ln = A - N;
double Ls = A - S;

double pw = Math.Min(A, W)/(Math.Abs(A - W)+ eps);
double pe = Math.Min(A, E)/(Math.Abs(A - E)+ eps);
double pn = Math.Min(A, N)/(Math.Abs(A - N)+ eps);
double ps = Math.Min(A, S)/(Math.Abs(A - S)+ eps);

        tmp = A + smooth*(Ln * pn + Ls * ps + Le * pe + Lw *
pw);

f[y, x]=new Gray( tmp);

if(first){ min= max = tmp; first =false;}
else
{
if(tmp < min) min = tmp;
else
        max = tmp;
}
}
}
}
return f;
}

#endregion

#region IsotropicSmoothing

publicstatic Image<Gray,byte>IsotropicSmoothing(Image<Gray,byte> im,int
smooth)
{
double eps =.1;
        Image<Gray,byte> f =new Image<Gray,byte>(im.Size.Width,
im.Size.Height);
bool first =true;
double tmp, min =0, max =0;
for(int y =0; y <f.Size.Height; y++)
{
for(int x =0; x <f.Size.Width; x++)
{
        tmp =im[y, x].Intensity;
if(x >0&& x <im.Size.Width-1&& y >0&& y < im.Size.Height -1)
{
double A = tmp;
double E =im[y +1, x].Intensity;
double S =im[y, x +1].Intensity;
double N =im[y, x -1].Intensity;
double W =im[y -1, x].Intensity;

double Lw = A - W;
double Le = A - E;
double Ln = A - N;
double Ls = A - S;

double pw = Math.Min(A, W)/(Math.Abs(A - W)+ eps);
double pe = Math.Min(A, E)/(Math.Abs(A - E)+ eps);
double pn = Math.Min(A, N)/(Math.Abs(A - N)+ eps);

```

```

double ps = Math.Min(A, S)/(Math.Abs(A - S)+ eps);

        tmp = A + smooth *(Ln + Ls + Le + Lw);

f[y, x]=new Gray(tmp);

if(first){ min= max = tmp; first =false;}
else
{
if(tmp < min) min = tmp;
else
                                max = tmp;
}
}
}
}
return f;
}

#endregion

publicstatic Image<Gray,byte>GrayHCH(Image<Gray,byte> im)
{
int x = Convert.ToInt32(im.Width*0.1);
int y = Convert.ToInt32(im.Height*0.1);

for(int j=0; j<im.Height; j++)
for(int i =0; i <im.Width; i++)
{
if(j >im.Height/2- y /2&& j < im.Height /2+ y /2&& i > im.Width /2- x /2&& i
< im.Height /2+ y /2)
continue;
im[j, i]=new Gray(0);
}
return im;
}

}
}

```