

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа информационных технологий и робототехники
 Направление подготовки 09.04.01 Информатика и вычислительная техника
 Отделение информационных технологий

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы Разработка алгоритмов генерации панорамных изображений из серии перекрывающихся кадров сцены

УДК 004.421.2:004.925.84:528.9

Студент

Группа	ФИО	Подпись	Дата
8ВМ91	Унжакова Екатерина Викторовна		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ, ИШИТР	Друки Алексей Алексеевич	к.т.н.		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН, ШБИП	Верховская Марина Витальевна	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД, ШБИП	Антоневич Ольга Алексеевна	к.б.н.		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ, ИШИТР	Кочегурова Елена Алексеевна	к.т.н.		

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП

Код компетенции	Наименование компетенции
	Универсальные компетенции
УК(У)-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий
УК(У)-2	Способен управлять проектом на всех этапах его жизненного цикла
УК(У)-3	Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели
УК(У)-4	Способен применять современные коммуникативные технологии, в том числе на иностранном (-ых) языке (-ах), для академического и профессионального взаимодействия
УК(У)-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия
УК(У)-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки
	Общепрофессиональные компетенции
ОПК(У)-1	Способен самостоятельно приобретать, развивать и применять математические, естественно-научные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте
ОПК(У)-2	Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач
ОПК(У)-3	Способен анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями
ОПК(У)-4	Способен применять на практике новые научные принципы и методы исследований
ОПК(У)-5	Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем
ОПК(У)-6	Способен разрабатывать компоненты программно-аппаратных комплексов обработки информации и автоматизированного проектирования
ОПК(У)-7	Способен адаптировать зарубежные комплексы обработки информации и автоматизированного проектирования к нуждам отечественных предприятий
ОПК(У)-8	Способен осуществлять эффективное управление разработкой программных средств и проектов
	Профессиональные компетенции
ПК(У)-1	Способен разрабатывать и администрировать системы управления базами данных
ПК(У)-2	Способен проектировать сложные пользовательские интерфейсы
ПК(У)-3	Способен управлять процессами и проектами по созданию (модификации) информационных ресурсов
ПК(У)-4	Способен осуществлять руководство разработкой комплексных проектов на всех стадиях и этапах выполнения работ
ПК(У)-5	Способен проектировать и организовывать учебный процесс по образовательным программам с использованием современных образовательных технологий

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа информационных технологий и робототехники
 Направление подготовки 09.04.01 Информатика и вычислительная техника
 Отделение информационных технологий

УТВЕРЖДАЮ:
 Руководитель ООП
 _____ Кочегурова Е.А.
 (Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

В форме:

Магистерской диссертации

Студенту:

Группа	ФИО
8ВМ91	Унжакова Екатерина Викторовна

Тема работы:

Разработка алгоритмов генерации панорамных изображений из серии перекрывающихся кадров сцены	
Утверждена приказом директора (дата, номер)	№34-35/с от 03.02.2021 г.

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

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	Объектом исследования является задача создания панорамных изображений. Разработанная и реализованная программная система способна сшивать панорамные изображения и обладает значительной гибкостью параметров и настроек, а её архитектура может быть достаточно легко расширена. Это делает её подходящей как для использования в качестве основы прикладного программного обеспечения, так и в качестве основы для дальнейшей исследовательской и экспериментальной работы, направленной непосредственно на совершенствования алгоритма создания панорамных изображений.
---------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Перечень подлежащих исследованию, проектированию и разработке вопросов	Обзор алгоритмов обработки и анализа изображений, проецирование изображений и получение бесшовной панорамы; проектирование структуры и содержания основных модулей разрабатываемой программной системы, разработка основных компонентов, визуализация панорамы с использованием веб-технологий; расчет ресурсоэффективности и ресурсосбережения, анализ вредных производственных факторов.
Перечень графического материала	Блок-схема алгоритма совмещения изображений, блок-схема сшивки панорамы, структура основных модулей программной системы, результаты панорамных изображений, изображений работы алгоритмов нахождения особых точек, сопоставление, схема веб-визуализации панорамы.

Консультанты по разделам выпускной квалификационной работы	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Доцент ОСГН, ШБИП, к. э. н., Верховская Марина Витальевна
Социальная ответственность	Доцент ООД, ШБИП, к. б. н., Антоневиц Ольга Алексеевна
Раздел на иностранном языке	Доцент ОИЯ, ШБИП, к. филол. н., Аксёнова Наталья Валерьевна

Названия разделов, которые должны быть написаны на русском и иностранном языках:
Введение, анализ предметной области, обзор алгоритмов и методов для создания панорамных изображений, разработка и реализация алгоритмов генерации панорамных изображений из серии перекрывающихся кадров сцены, результаты экспериментов, финансовый менеджмент, ресурсоэффективность и ресурсосбережение, социальная ответственность на русском языке.
Обзор алгоритмов и методов для создания панорамных изображений, разработка и реализация алгоритмов генерации панорамных изображений из серии перекрывающихся кадров сцены на английском языке.

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	01.03.2021
-------------------------------------------------------------------------------------------------	------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОИТ ИШИТР	Друки Алексей Алексеевич	к.т.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ВМ91	Унжакова Екатерина Викторовна		

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

Студенту:

Группа	ФИО
8ВМ91	Унжакова Екатерина Викторовна

Школа	ИШИТР	Отделение школы (НОЦ)	ОИТ
Уровень образования	магистратура	Направление/специальность	09.04.01 «Информатика и вычислительная техника»

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

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Оклад инженера – 19254 руб. Оклад руководителя – 35111,5 руб.
2. <i>Нормы и нормативы расходования ресурсов</i>	Районный коэффициент 30%; Коэффициент дополнительной заработной платы 12%; Накладные расходы 16%.
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Коэффициент отчислений на уплату во внебюджетные фонды 30%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого и инновационного потенциала НТИ</i>	- разработка графика Гантта.
2. <i>Разработка устава научно-технического проекта</i>	-Анализ конкурентных технических решений
3. <i>Планирование процесса управления НТИ: структура и график поведения, бюджет, риски и организация закупок</i>	-Накладные расходы.
4. <i>Определение ресурсной, финансовой, экономической эффективности</i>	-Определение потенциального эффекта исследования.

Перечень графического материала:

1. <i>«Портрет» потребителя результатов НТИ</i>
2. <i>Сегментирование рынка</i>
3. <i>Оценочная карта конкурентных технических решений</i>
4. <i>Диаграмма Исикавы</i>
5. <i>Матрица SWOT</i>
6. <i>Оценка готовности проекта к коммерциализации</i>
7. <i>График проведения и бюджет НТИ</i>
8. <i>Оценка ресурсной, финансовой и экономической эффективности НТИ</i>

Дата выдачи задания для раздела по линейному графику	01.03.2021
-------------------------------------------------------------	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН, ШБИП	Верховская Марина Витальевна	к.э.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ВМ91	Унжакова Екатерина Викторовна.		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8ВМ91	Унжакова Екатерина Викторовна

Школа	ИШИТР	Отделение (НОЦ)	Информационных технологий
Уровень образования	Магистратура	Направление/специальность	09.04.01 «Информатика и вычислительная техника»

Тема ВКР:

Разработка алгоритмов генерации панорамных изображений из серии перекрывающихся кадров сцены	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объект исследования: задача генерации панорамного изображения. Область применения: создание трехмерных карт местности и панорамных изображений, может быть применено в сферах туризма, картографии, космической разработке и т.д. Рабочая зона: постоянное рабочее место с персональной электронно-вычислительной машиной (ПЭВМ).
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности: <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<ul style="list-style-type: none"> – СП 2.2.3670-20 «Санитарно-эпидемиологические требования к условиям труда». – ГОСТ 12.2.032-78 «Система стандартов безопасности труда. Рабочее место при выполнении работ сидя». – ГОСТ 22269-76 «Рабочее место оператора. Взаимное расположение элементов рабочего места». – ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения».
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	Вредные факторы: <ul style="list-style-type: none"> – монотонный режим работы; – нервно-психические перегрузки, связанные с напряженностью трудового процесса; – отклонение показателей микроклимата; – повышенный уровень шума; – недостаточная освещенность рабочей зоны. Опасные факторы:

	– повышенное значение напряжения в электрической цепи, замыкание которой может пройти через тело человека.
3. Экологическая безопасность:	Программное обеспечение не связано с производством, следовательно, влияние на окружающую среду минимально. Литосфера: рассмотрена утилизация бумажных отходов и неисправных комплектующих.
4. Безопасность в чрезвычайных ситуациях:	Возможные ЧС: пожары и взрывы. Наиболее типичная ЧС: пожары.

Дата выдачи задания для раздела по линейному графику	01.03.2021
------------------------------------------------------	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД, ШБИП	Антоневич Ольга Алексеевна	к.б.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ВМ91	Унжакова Екатерина Викторовна		

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа информационных технологий и робототехники
 Направление подготовки 09.04.01 Информатика и вычислительная техника
 Уровень образования: магистр
 Отделение информационных технологий
 Период выполнения: осенний / весенний семестр 2020 /2021 учебного года

Форма представления работы:

Магистерская диссертация

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

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

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
22.05.2021	<i>Основная часть</i>	70
25.05.2021	<i>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</i>	10
25.05.2021	<i>Социальная ответственность</i>	10
25.05.2021	<i>Обязательное приложение на иностранном языке</i>	10

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ, ИШИТР	Друки Алексей Алексеевич	к.т.н.		

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ, ИШИТР	Кочегурова Елена Алексеевна	к.т.н.		

Реферат

Выпускная квалификационная работа содержит 136 страниц, 39 рисунков, 40 таблиц, 51 источник литературы, 4 приложения.

Ключевые слова: создание панорамных изображений, ключевые особенности, детектирование особенностей, бесшовная панорама, преобразования перспективы, гомография, аффинные преобразования.

Объектом исследования является задача создания бесшовного панорамного изображения.

Цель работы – разработка и реализация системы алгоритмов для создания панорамных изображений из наборов фотографий, а также исследование эффективности в рамках этой цели различных методов и параметров извлечения и сравнения особенностей (ключевых точек) изображения.

В процессе исследования проводился обзор алгоритмов и методов для создания панорамных изображений, проанализированы метрики сопоставления ключевых особенностей, поиск наилучшего перспективного преобразования и формы проекции для веб-реализации и сшивка панорамных изображений.

В результате исследования были предложены и разработаны несколько алгоритмов: совмещение изображений с получением статистических данных, сортировка изображений, смещение изображений, цветовая коррекция, сшивка пары изображений, цилиндрическая проекция, создание полноценной панорамы, а также визуализация готовой панорамы с использованием веб-технологий.

Область применения: использование в качестве основы прикладного программного обеспечения для реализации приложений, основанных на компьютерном зрении в самых различных областях применения, например, виртуальные туры, аэрофотосъемка и картографирование и др., так и в качестве основы для дальнейшей исследовательской и экспериментальной работы, направленной непосредственно на совершенствования алгоритма создания панорамных изображений.

Обозначения и сокращения

OpenCV (Open Source Computer Vision Library) – библиотека компьютерного зрения с открытым исходным кодом.

SIFT (Scale-Invariant Feature Transform) – масштабно-инвариантное преобразование особенностей.

SURF (Speeded-Up Robust Features) – ускоренные устойчивые преобразования признаков.

ORB (Oriented FAST and Rotated BRIEF) – ориентированный FAST, полученный из ускоренной проверки сегментов и дескриптор BRIEF.

FAST (Features from Accelerated segment test) – признаки, полученные из ускоренной проверки сегментов.

BRIEF – быстрый эвристический дескриптор, строится из 256 бинарных сравнений между яркостями пикселей на размытом изображении.

BRISK (Binary Robust Invariant Scalable Keypoints) – бинарные устойчивые инвариантные масштабируемые ключевые точки.

BFMatcher (Basics of Brute-Force Matcher) – сопоставитель методом грубой силы.

FlannBasedMatcher (Fast Library for Approximate Nearest Neighbors based Matcher) – быстрая библиотека для приблизительных ближайших соседей.

LSH (Locality-sensitive hashing) – вероятностный метод понижения размерности многомерных данных.

RANSAC (Random Sample Consensus) – стабильный метод оценки параметров модели на основе случайных выборок.

LMEDS (Least-median of Squares) – метод наименьших полных квадратов, или алгоритм наименьшей медианы квадратов.

PROSAC (Progressive Sample Consensus) – прогрессивный метод однородной выборки.

Оглавление

Введение.....	13
1 Анализ предметной области	14
1.1 Актуальность разработки	14
1.2 Получение качественных картинок и рекомендации съемки.....	15
2 Обзор алгоритмов и методов для создания панорамных изображений	18
2.1 Общая концепция детекции особенностей.....	18
2.2 Детектирование особенностей.....	19
2.3 Соответствие особенностей	22
2.4 Вычисление матрицы гомографии	25
2.5 Проекция панорамных изображений	28
2.6 Получение бесшовной панорамы	33
3 Разработка и реализация алгоритмов генерации панорамных изображений из серии перекрывающихся кадров сцены	38
3.1 Функциональные и нефункциональные требования.....	38
3.2 Средства реализации программной системы	39
3.3 Общая программная система алгоритмов для создания панорамного изображения.....	41
3.4 Разработка и реализация визуализации панорамного изображения.....	53
4 Результаты экспериментов.....	55
4.1 Сравнительный анализ полученных результатов эффективности методов поиска ключевых точек	55
4.2 Сравнительная демонстрация сшивки изображений с использованием цветокоррекции и смещения.....	59
4.3 Сравнение с другими программными системами для сшивки панорамных изображений	61
5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение ...	67
5.1 Предпроектный анализ	67
5.2 Планирование управления научно-техническим проектом.....	76
5.3 Оценка сравнительной эффективности исследования	82

6	Социальная ответственность	86
6.1	Правовые и организационные вопросы обеспечения безопасности	86
6.2	Производственная безопасность	90
6.3	Экологическая безопасность.....	100
6.4	Безопасность в чрезвычайных ситуациях.....	101
	Заключение	105
	Список публикаций студента.....	107
	Список используемых источников.....	108
	Приложение А Результаты алгоритма для совмещения изображений и получения статистических данных	113
	Приложение Б Результаты сопоставления с использованием вероятностных методов	120
	Приложение В Результаты сшивки панорам в различных системах.....	121
	Приложение Г Development of algorithms for creating panoramic images.....	124

Введение

Компьютерное зрение и обработка изображений – это широкая, важная и востребованная в настоящее время как прикладная область, так и фундаментальное научное направление. Как научная дисциплина, компьютерное зрение относится к теории и технологии создания искусственных систем, которые получают информацию из изображений. Как технологическая дисциплина, компьютерное зрение стремится применить свои теории и модели к созданию программно-аппаратных систем.

Целью данной работы является разработка и реализация алгоритмов для создания панорамных изображений из наборов фотографий, а также исследование эффективности в рамках этой задачи различных методов и параметров извлечения и сравнения особенностей (ключевых точек) изображения.

Объектом исследования является изображение, а предметом исследования – алгоритм генерации изображений в панораму. Предметом разработки является программная система для создания панорамных изображений.

Разработанная и реализованная программная система способна сшивать панорамные изображения и обладает значительной гибкостью параметров и настроек, а её архитектура может быть достаточно легко расширена. Это делает её подходящей как для использования в качестве основы прикладного программного обеспечения, так и в качестве основы для дальнейшей исследовательской и экспериментальной работы, направленной непосредственно на совершенствования алгоритма создания панорамных изображений.

1 Анализ предметной области

1.1 Актуальность разработки

Существуют несколько подходов к созданию панорамных снимков: техническое решение, основанное на использовании видеопанорамных камер и автоматическая сшивка в приложении, и программное решение, заключающееся в обработке исходных кадров с перекрывающимися областями, сшиваемых в один панорамный снимок.

Техническое решение с использованием видеопанорамной камеры в комплексе со штативом, как единая система, может быть весьма дорогостоящим и нуждается в обслуживании и калибровке. При использовании приложений с автоматической сшивкой, отсутствует возможность заменить кадр общего снимка и необходимо переснимать всю панораму в целом. При использовании готового программного решения пользователи лишены возможности менять параметры сшивки.

Для создания панорамных снимков из серии перекрывающихся кадров сцены используется подход, основанный на поиске соответствий между изображениями со сбором статистических данных и вычислении перспективных преобразований цилиндрической формы.

Актуальность разработки программной системы обусловлена независимостью от использования дорогостоящих видеосистем и возможностью получения статистических данных в процессе сшивки изображений для научно-исследовательских работ.

1.2 Получение качественных картинок и рекомендации съемки

Панорама – это такое изображение, которое собрано из серии последовательно снятых фотографий. Панорамные снимки позволяют добиться большого угла обзора и получать изображения огромного разрешения.

Как правило, чаще всего процесс съемки панорамного пейзажа представляет собой снимки нескольких кадров при повороте камеры в новое положение. Для получения качественных, без следов сшивки, изображений, снимать кадры для панорамы лучше с использованием штатива и специальной панорамной головки, а повороты производить на один и тот же градус.

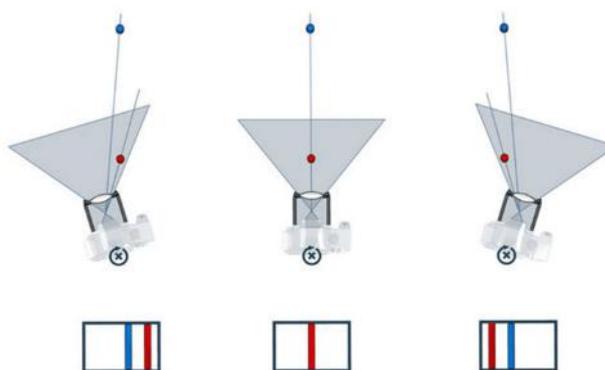
Камера вертикально устанавливается на панорамную головку, которая позволяет вращать камеру вокруг беспараллаксной точки (рис. 1).



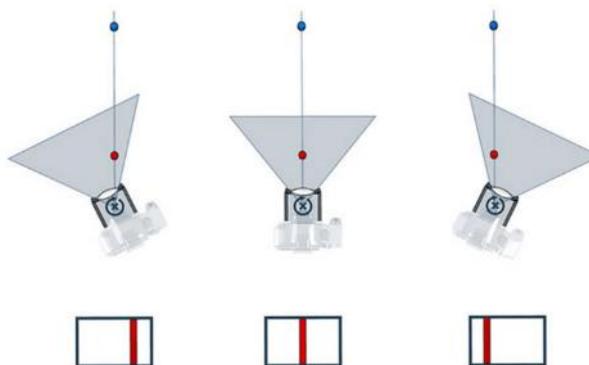
Рис. 1 – Положение камеры с использованием штатива

Поскольку часто не удастся сшить отснятых с рук снимки в единую панораму, то рекомендуется использовать штатив. Причиной этого является эффект «параллакс» – изменение видимого положения объекта относительно удалённого фона в зависимости от положения наблюдателя.

Например, на рисунке 2 видно, что при повороте камеры происходит смещение объектов ближнего и дальнего плана друг относительно друга. Чтобы этого избежать, необходимо вращать фотоаппарат вокруг специальной точки, которая называется нодальной. При вращении камеры вокруг нодальной точки смещения объектов ближнего и дальнего плана уже не происходит [1].



а) смещение объектов ближнего и дальнего плана



б) отсутствие смещения при вращении вокруг нодальной точки

Рис. 2 – Влияние параллакса на снимок при повороте камеры

Для вращения фотоаппарата непосредственно вокруг нодальной точки применяются панорамные головки, которые накручиваются на штатив и тем самым позволяют вращать фотоаппарат на 360 градусов. Определённые модели панорамных головок могут быть приспособлены исключительно для съёмки цилиндрических панорам (т. е. фотоаппарат можно вращать только по горизонтали). Однако большая часть головок позволяют создавать как сферические, так и цилиндрические панорамы. В таком случае вращение фотоаппарата возможно и по горизонтали, и по вертикали (рис. 3).

Стоит также упомянуть про процент перекрытия кадров. Перекрытие кадров является критически важным для сшивки панорамных изображений. Оно делается для того, чтобы алгоритм смог выявить контрольные точки – т. е. определить характерные особенности для соседних кадров. Таким образом, значение перекрытия непосредственно зависит от детализации и особенностей снимаемой сцены.

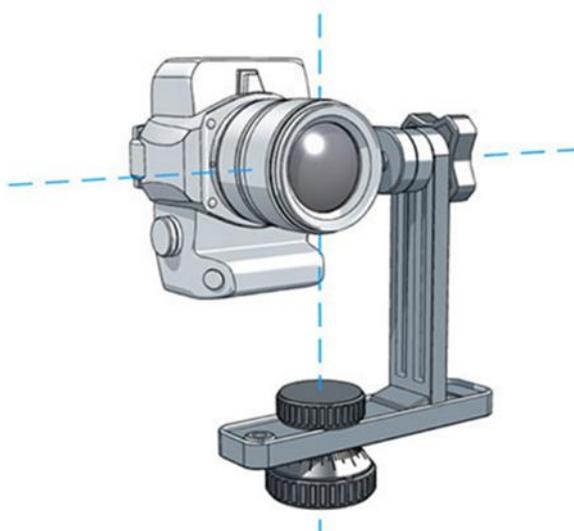


Рис. 3 – Возможность вращения фотоаппарата вдоль нодальной точки

К примеру, если снимается единственный крупный объект, то в этом случае нужно сделать большее (до 80%) перекрытие, следя за тем, чтобы наиболее заметные детали попадали в это перекрытие, а если снимается пейзаж, который имеет множество деталей в кадре, то следует делать процент перекрытия в пределах 10-30%, но обычно среднестатистическим процентом перекрытия для уверенной сшивки панорамы следует считать значения в диапазоне от 20 до 40%, (рис. 4).



Рис. 4 – Демонстрация варианта перекрытия двух кадров общей сцены

Необходимо отметить, что процент перекрытия кадров определяет их количество в круговой панораме, а само количество кадров для сшивки будет влиять на скорость работы алгоритма сшивки и на требуемые вычислительные мощности.

2 Обзор алгоритмов и методов для создания панорамных изображений

2.1 Общая концепция детекции особенностей

В компьютерном зрении существует важное направление – детекция особенностей (feature detection) на изображениях, из которых, собственно, ключевые точки и определяются как таковые. Под этим можно подразумевать не просто точки, как пару координат, а некоторые области на изображениях, для нахождения которых и существуют различные методы и алгоритмы.

Собственно особенности изображения могут быть различных видов [2].

- Рёбра (edges) – границы между двумя областями изображений, это могут быть случайные очертания и стыки разных объектов; на практике часто набор точек, которые имеют большую величину градиента.

- Углы / точки интереса, ключевые точки (corners / interest points, фактически синонимы). Изначально продукт анализа рёбер с целью нахождения резких изменений в направлении (т.е. углов), затем были разработаны алгоритмы, не требующие явного вычисления рёбер (например, ищущие сильное искривление градиента изображения). Отметим, что найденные ключевые точки совершенно не обязательно являются углами объектов сцен в прямом смысле, но тем не менее они — особенности, которые учитываются.

- Сферы / ключевые области, ключевых точек (blobs / regions of interest or interest points). Фактически, описание областей изображение. Также как и углы считаются ключевыми точками (и имеют фактический центр), однако у них свои алгоритмы расчётов.

Различные алгоритмы нахождения особенностей всегда схожи в одном – как только особенность найдена, происходит извлечение (extraction) небольшой области вокруг неё. Результат – дескриптор или вектор особенности. Это относится и к углам и к сферам. Различные алгоритмы поиска особенностей дают различные дескрипторы.

Методы сопоставления дескрипторов (matchers) ключевых точек на разных изображениях – краеугольный камень нахождения шаблонов на сцене, возможность компьютерного зрения для решения задачи распознавание образов.

И эти способы сопоставления могут быть отличными для различных алгоритмов поиска ключевых точек и различных дескрипторов.

Существуют различные подходы (и их реализации) к нахождению ключевых точек, это как раз методы, рассматриваемые ниже.

2.2 Детектирование особенностей

Для выделения ключевых точек существует несколько основных методов, уже реализованных в различных программных средствах и широко используемых на практике.

Методы как минимум включают в себя алгоритмы поиска, возвращающие список ключевых точек, а также обычно (но не всегда) имеют встроенные алгоритмы извлечения дескрипторов. Если встроенных алгоритмов извлечения нет, то существуют реализованные отдельно, способные извлекать из изображения дескрипторы определённых форматов по найденным ключевым точкам.

Основные методы поиска ключевых точек на изображениях, имеющих доступную, готовую для использования программную реализацию, это SIFT, SURF, ORB, BRISK и AKAZE. Алгоритмически это различные методы, тяготеющие к разным особенностям изображений.

Эти методы поиска ключевых точек наиболее популярные и распространённые (есть и другие, обычно модификации указанных), кроме того важно то, что их реализации имеются в одной открытой библиотеке компьютерного зрения OpenCV, что делает неактуальным их реализацию с нуля (либо использование разных источников/библиотек) и даёт возможность сравнивать их работу в идентичных условиях.

Соответственно, данная выпускная квалификационная работа фокусируется на экспериментальном сравнении метрик для сшивки изображений именно этих методов.

Таким образом, их сравнение позволит рассуждать о целесообразности применения методов для реальной задачи, их конкурентоспособность друг относительно друга.

Очевидно, предназначенная для экспериментов система должна включать в себя реализацию данных методов. Далее – краткое описание каждого из них.

2.2.1 Метод SIFT

SIFT – старейший (опубликован в 1999г.) из выбранных методов, но пользующийся немалой популярностью. Предоставляет алгоритмы поиска ключевых точек посредством вычисления сфер (blobs) и извлечения дескрипторов.

Ключевые точки здесь находятся при помощи вычисления экстремума масштаба-пространства (Scale-space) – изображение свёртывается фильтрами Гаусса в разных масштабах и затем берётся разность последовательных, «размытых» методом Гаусса изображений, а ключевые точки берутся как максимум/минимум разницы гауссианов, которая имеет место на нескольких масштабах. Полученных таким образом «кандидатов» в ключевые точки очень много и некоторые могут быть нестабильны, поэтому затем проводится локализация с отсеиванием нестабильных (на этом этапе также задействованы и углы, edges), а уже затем возможно извлечение дескрипторов [3].

2.2.2 Метод SURF

SURF – метод, вдохновлённый SIFT. Кратко суть метода можно расписать следующим образом. Ключевые точки (фактически – участки) определяются максимальным изменением градиента яркости. Для этого используются матрица Гессе (детерминант – гессиан – достигает экстремума в точках максимального изменения градиента яркости) и интегральное представление изображений (как раз позволяет быстро считать суммы яркостей). После нахождения для точек извлекаются дескрипторы, обеспечивающие инвариантность масштаба и вращения. Таким образом, потенциально обеспечивается нахождение одних и тех же точек на изображениях с разным ракурсом/масштабом сцены [4][5]. Можно сказать, что этот метод находя ключевые точки-сферы (blobs) и умея извлекать для них дескрипторы имеет архитектуру подобную SIFT.

В ходе исследования было выявлено, что метод SURF деактивирован в актуальной версии библиотеки OpenCV по причине того, что он запатентован, поэтому не предоставляется возможным его использование.

2.2.3 Метод ORB

ORB – самый новый из выбранных методов (2011г.). Как следует из развёрнутого названия, это слияние метода FAST и дескриптора BRIEF [6], подвергнутых модификации. Сначала методом FAST находятся точки, затем применяя меру угла Харриса находятся N верхних точек среди них. Также используется пирамида для получения мультимасштабных особенностей. Далее (в отличие от FAST) здесь реализовано вычисление ориентации найденных точек с помощью вычисления интенсивности взвешенного центроида от лоскута с найденным углом в центре, и направление от точки угла до центроида даёт ориентацию. Для улучшения инвариантности поворота, моменты вычисляются для x и y которые внутри радиуса r (размер лоскута). Далее работает дескриптор BRIEF. По причине плохой работы дескриптора с ориентацией, ORB будет направлять его, строя матрицы поворота для ключевых точек.

ORB считается более быстрым, чем SIFT и SURF, и его дескрипторы работают лучше, чем у SURF [7].

2.2.4 Метод BRISK

BRISK так же, как и ORB, создан на основе комбинации детектора FAST с оптимизированным дескриптором BRIEF, однако в их работу внесены изменения. Для обеспечения инвариантности метода к масштабированию, также используется пирамида Гаусса, в которой производится поиск особых точек с наилучшей максимальной интенсивностью в пирамиде при помощи детектора FAST [8]. Для поиска угловых точек (corners) поочерёдно рассматриваются окрестности по 16 пикселей вокруг каждого пикселя.

Дескриптор направленный BRIEF представляется в виде вектора длиной 256, состоящего из результатов бинарных тестов вокруг особой точки. В

окрестности 31×31 пиксель сравниваются средние значения яркостей между x и y , где x, y – области 5×5 пикселей.

2.2.5 Метод АКАZE

АКАZE (Accelerated-KAZE) является усовершенствованной версией KAZE путем применения алгоритма FED (Fast Explicit Diffusion) [9] для построения нелинейной многомасштабной пирамиды, который работает быстрее благодаря относительной простоте реализации. Применение нелинейного коэффициента масштабирования позволяет увеличить скорость нахождения нужной особой точки по сравнению с Гауссовой пирамидой. В качестве дескриптора АКАZE использует дескриптор M-LDB инвариантный к вращению и масштабированию. В M-LDB тесты проводятся не между средним значением всех пикселей в области, а между заданным их количеством в зависимости от размера, что позволяет ускорить вычисление дескриптора.

2.3 Соответствие особенностей

После того, как были обнаружены ключевые особенности изображений, необходимо выбрать алгоритм для сопоставления их характеристик.

k-means (метод k-средних) – метод кластеризации, производящий разбиение некоторой заданной выборки объектов (данных) на непересекающиеся подмножества/группы (кластеры) на основе близости их признаков/значений, который стремится минимизировать суммарное квадратичное отклонение точек кластеров от центров этих кластеров [10]:

$$V = \sum_{i=1}^k \sum_{x \in S_i} (x - \mu_i)^2 \quad (1)$$

где k – число кластеров, S_i – полученные кластеры, $i = 1, 2, \dots, k$, а μ_i – центры масс всех векторов x из кластера S_i .

Таким образом, каждый кластер состоит из схожих между собой объектов.

Методу k-средних часто отдается предпочтение благодаря простоте его реализации и значительной скорости работы. Действие данного алгоритма заключается в стремлении минимизировать суммарное квадратичное

отклонение точек кластеров от центров этих кластеров. Говоря иначе, это алгоритм, который итеративно делит данное множество пикселей на k кластеров точек, которые являются максимально приближенными к их центрам, а сама кластеризация происходит за счет смещения этих же центров. Стоит также отметить то, что метод k -средних весьма чувствителен к шуму, который может значительно исказить результаты кластеризации. Метод эффективен в случае, если данные делятся на компактные группы, которые можно описать сферой.

Далее рассмотрим некоторые кластерные методы, которые основаны на мере расстояния.

Реализация метода k -средних с использованием метрики расстояния Евклида сравнивает каждый набор дескрипторов из одного изображения с каждым таким набором из другого изображения и в результате находит наилучшее соответствие между особыми точками. Расстояние Евклида вычисляется по классической формуле [11]:

$$D = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (2)$$

где D – расстояние между объектами, n – количество характеристик, p и q – сравниваемые векторы с i – свойством.

Метод, основанный на теореме Пифагора, прост в использовании полученных результатов, возвращает не гарантированно верное решение, а лучшее из возможных. Данный метод может быть затратным по части вычислительных ресурсов, поскольку при его использовании может потребоваться полный перебор найденных ключевых особенностей.

Манхэттенское расстояние – это расстояние, являющееся средним для разностей по координатам [12]. Чаще всего, данная мера расстояния приводит к практически таким же результатам, как и для ранее рассмотренного расстояния Евклида. Тем не менее, особенностью такой меры является уменьшение влияние отдельных больших разностей (выбросов), т.к. они не возводятся в квадрат:

$$D = \sum_{i=1}^n (x_i - y_i)^2 \quad (3)$$

Расстояние Хэмминга – это число отличающихся бит в двух сравниваемых бинарных векторах (единственная допустимая операция – замена)

[11] одинаковой длины. Является метрикой различия (функцией, которая определяет расстояние в метрическом пространстве) объектов одинаковой размерности:

$$D_{ij} = \sum_{s=1}^p |x_{is} - y_{js}| \quad (4)$$

где p – размерность, x и y – двоичные последовательности (вектора) длины n .

Расстояние Хэмминга просто реализуется и быстро работает (т. е. имеет высокую производительность) и на данный момент достаточно широко используется для разнообразных задач, например поиска близких дубликатов, распознавания образов, классификации документов, исправления ошибок, обнаружения вирусов и т.д. У расстояния Хэмминга также есть заметный недостаток, а именно требование одинаковой длины строк. Таким образом, данная метрика оптимальна для расчета расстояния редактирования с учетом таких искажений, как замена и транспозиция, однако не подходит при вставках и удалениях.

LSH – вероятностный метод понижения размерности многомерных данных. Его идеей является подбор хеш-функций для некоторых измерений таким образом, чтобы похожие объекты с высокой степенью вероятности попадали в одну условную «корзину» [13]. Данный метод позволяет строить структуру для быстрого приближенного (вероятностного) поиска n -мерных векторов, «похожих» на искомый шаблон. Среди имеющихся на сегодня приближенных алгоритмов поиска ближайших соседей именно LSH является, вероятно, наиболее популярным.

KD-Tree (K-мерное дерево) [14], специальная структура данных, которая позволяет разбить K-мерное пространство на «меньшие части», посредством сечения этого самого пространства гиперплоскостями ($K > 3$), плоскостями ($K = 3$), прямыми ($K = 2$), и в случае одномерного пространства-точкой (выполняя поиск в таком дереве, получается что-то похожее на бинарный поиск). Логично, что такое разбиение обычно используют для сужения диапазона поиска в K-мерном пространстве. Например, поиск ближнего объекта (вершины, сферы,

треугольника и т.д.) к точке, проецирование точек на 3D сетку, трассировка лучей (активно используется в Ray Tracing) и т.п. При этом объекты пространства помещаются в специальные параллелепипеды – bounding box-ы (bounding box-ом назовем такой параллелепипед, который описывает исходное множество объектов или сам объект, если мы строим bounding box лишь для него. У точек в качестве bounding box-а берется bounding box с нулевой площадью поверхности и нулевым объемом), стороны которых параллельны осям координат.

2.4 Вычисление матрицы гомографии

Гомографией называется обратимое преобразование некоторой реальной проективной плоскости на проективную плоскость, переводящее прямые линии в прямые линии. Её можно представить в виде матрицы 3×3 с восемью степенями свободы:

$$H = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix}, \tilde{X} \propto HX \quad (1)$$

где H – матрица гомографии, $X = [x, y, 1]^T$ – произвольный пиксель исходного изображения, $\tilde{X} = [\tilde{x}, \tilde{y}, \tilde{w}]^T$ – координаты пикселя после проецирования. После перехода от однородных координат результатом будут новые координаты (\hat{x}, \hat{y}) исходной точки изображения (x, y) , что в общем виде можно записать следующим образом:

$$\begin{cases} \hat{x} = \frac{ax+by+c}{gx+hy+1} \\ \hat{y} = \frac{dx+ey+f}{gx+hy+1} \end{cases} \quad (2)$$

Гомографическая матрица H определяется восемью элементами, из которых a и e отвечают за масштабирование координат по осям OX и OY соответственно, b и d – за сдвиг по осям OX и OY (вместе с a , e влияют на поворот), c и f – за смещение по осям OX и OY , а g и h – на изменение перспективы.

Матрица гомографии определяет перспективное преобразование второго изображения по отношению к первому, эталонному, изображению. Далее на

основе полученной функции преобразования производится трансформация изображения для выравнивания второго изображения по отношению к первому. Уже трансформированная версия второго изображения в свою очередь накладывается перед эталонным изображением, т.е. происходит сшивка изображений.

Преобразование перспективы – это наиболее сложная форма двумерных преобразований плоскости. По части возрастания сложности, преобразования можно расположить в следующем порядке: перенос, Евклидово преобразование, преобразование подобия, аффинное преобразование и перспективное преобразование [15] (рис. 5). В случае если гомография вычисляется в небольшой области изображения или же изображение было получено с большим фокусным расстоянием, то аффинная гомография является более оптимальной моделью сшивки изображения.

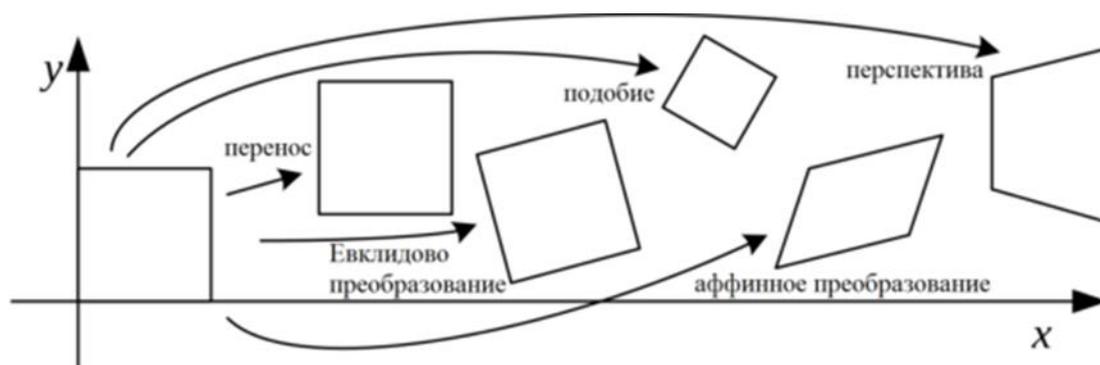


Рис. 5 – Двумерные преобразования плоскости

В выпускной квалификационной работе было решено реализовать алгоритм поэтапной сшивки изображений с определением матрицы гомографии и аффинными преобразованиями.

Неверных совпадений (outliers) невозможно избежать полностью на этапе сопоставления признаков, таким образом для точной подгонки модели преобразования требуется ещё дополнительная фаза исключения неподходящих точек.

После того, как начальный набор соответствий признаков был вычислен, нужно найти подмножество, которое обеспечит выравнивание с высокой точностью.

RANSAC [16], LMEDS [17] и PROSAC [18] – это некоторые из надежных вероятностных методов, которые могут быть использованы (и часто используются на практике) для удаления неверных совпадений из сопоставленных характеристик и подбора функции преобразования (в терминах матрицы гомографии).

2.4.1 Алгоритм RANSAC

RANSAC – это итерационный метод построения надежных оценок параметров математической модели на основе набора наблюдаемых данных, которые, как известно, содержат выбросы.

Этот алгоритм является недетерминированным в том смысле, что он дает приемлемый результат только с определенной вероятностью, причем сама эта вероятность возрастает с увеличением количества итераций.

Метод впервые был опубликован в 1981 году Фишлером и Боллесом. Его основное предположение состоит в том, что данные состоят из «выбросов», то есть данных, распределение которых можно объяснить с помощью некоторой математической модели, и «выбросов», которые представляют собой данные, не соответствующие модели. Ключевые точки, являющиеся «выбросами», могут рассматриваться как результаты, вызванные шумом, ошибочными измерениями или просто неверными данными. RANSAC также предполагает, что для данного набора наборов существует процедура, которая может оценить параметры модели, которая оптимально объясняет или соответствует этим данным. Метод RANSAC способен обрабатывать практически любое соотношение выбросов, однако ему необходим специальный параметр, порог, чтобы отличать «хорошие» ключевые точки от выбросов.

Алгоритм RANSAC часто используется в сфере компьютерного зрения, в частности для одновременного решения проблемы соответствия и оценки фундаментальной матрицы, относящейся к паре стереокамер.

2.4.2 Алгоритм LMEDS

LMEDS – это один из наиболее ранних типов устойчивой регрессии. Его можно назвать «регрессия методом наименьших квадратов», и идея его состоит в минимизации суммы квадратов остатков.

В частном случае простой регрессии это соответствует поиску самой узкой полосы, покрывающей половину наблюдений, которая уменьшает влияние остатков. Согласно Венейблсу и Рипли (1999), этот алгоритм минимизирует медианное значение упорядоченных квадратов остатков для получения коэффициента регрессии.

Для алгоритма LMEDS не требуется пороговое значение, однако он работает правильно только при наличии более 50% меток. Метод очень прост с вычислительной точки зрения. Оценки коэффициентов регрессии при определенных предпосылках обладают рядом оптимальных свойств.

2.4.3 Алгоритм PROSAC

Структура алгоритма PROSAC аналогична RANSAC. Сначала путем случайной выборки генерируются гипотезы. Выборки, в отличие от RANSAC, берутся из некоторого подмножества данных с наивысшим качеством, а не из всех данных. Размер набора для генерации гипотез постепенно увеличивается. Поэтому на раннем этапе исследуются именно те ключевые особенности, которые с большей вероятностью с наименьшим количеством шума.

Фактически, PROSAC предназначен для отрисовки тех же характерных особенностей, что и RANSAC, только в другом порядке. Гипотезы проверяются на соответствие всем данным. Как и в RANSAC, алгоритм завершается, когда вероятность существования решения, которое на данный момент было бы лучше, чем наилучшее, становится низкой (менее 5%).

2.5 Проекция панорамных изображений

После определения матриц гомографии всех входных изображений важным вопросом является выбор системы координат, пространства для размещения преобразованных изображений.

Начиная работу с панорамными изображениями, необходимо получить некоторое представление о различных существующих проекциях панорамных изображений и о том, как они влияют на изображения:

1. Прямолинейная проекция. Признавая, что окружающая среда вокруг вас принимает форму сферы, прямолинейная проекция используется, когда фотографируется только часть этой сферы (рис. 6). Затем эта часть сферы проецируется на плоскую поверхность. Часто используется для склейки небольшого количества изображений: хорошо адаптирована для частичных панорам и мозаичных изображений.



Рис. 6 – Прямолинейная проекция

2. Цилиндрическая проекция. Будет использоваться, когда на панораме нет данных о зените или надире (полюсах сферы) – обычно это однорядное изображение. Это проекция, используемая для цилиндрических панорам и некоторых более крупных частичных панорам, которые выглядят слишком искаженными в прямолинейной проекции. Он проецирует часть сферы (цилиндр) на плоскую поверхность (изображение) (рис. 7).

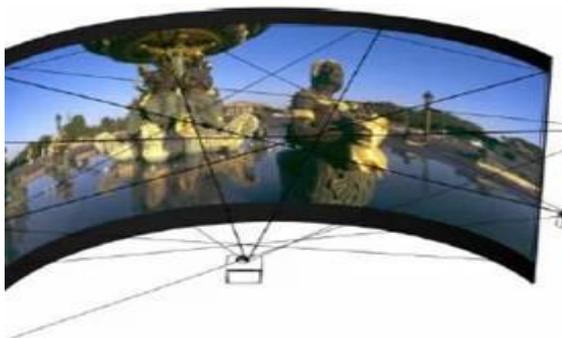


Рис. 7 – Цилиндрическая проекция

3. Сферическая (эквидистантная) проекция. Эта проекция используется для проецирования всей сферы на плоскую поверхность (рис. 8). В результате получается равнопрямоугольное изображение с соотношением сторон 2:1. Ширина ровно в два раза больше высоты. Это логично, так как охватывает 360° по горизонтали и 180° по вертикали. Можно увидеть, что в этой проекции верхняя и нижняя части изображения выглядят очень искаженными, и это нормально. Эта проекция, используется в качестве источника для полной сферической панорамы.



Рис. 8 – Сферическая проекция

4. Проекция Меркатора. Эту проекцию можно рассматривать как производную от равнопрямоугольной проекции без искажений в зените и надире (рис. 9). Карты мира находятся в этой проекции. Для панорам это может быть преимуществом, если нужно, например, распечатать изображение сферической панорамы или использовать изображение на веб-сайте. Но эту проекцию нельзя использовать для создания интерактивных панорам.



Рис. 9 – Проекция Меркатора

5. Кубическая проекция. Эта проекция используется как разбивка равнопрямоугольного изображения (сферической панорамы) (рис. 10). Принцип

состоит в том, что вся сфера превращается в 6 граней куба – переднюю, правую, заднюю, левую, зенит и надир. Каждое из 6 квадратных изображений имеет прямолинейную проекцию.



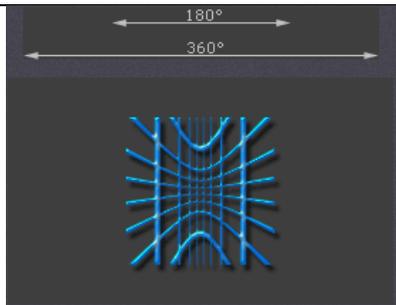
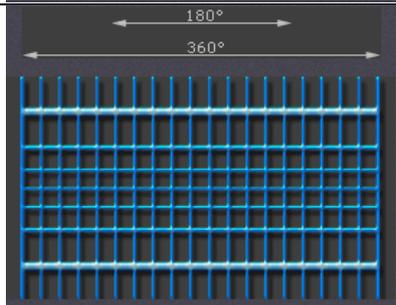
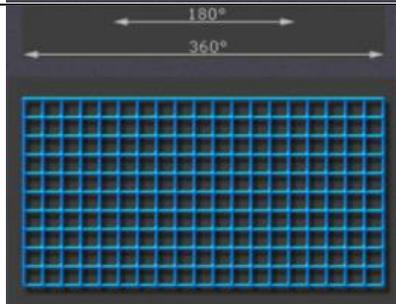
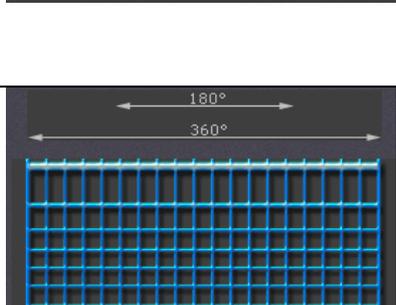
Рис. 10 – Кубическая проекция

Эта проекция очень полезна, когда нужно отредактировать изображение сферической панорамы. Не всегда легко редактировать деформированные изображения на равнопрямоугольном изображении, а также невозможно редактировать полностью искаженные зенит и надир. Разбив это искаженное изображение на 6 плоских изображений (прямолинейная проекция – это изображение с меньшим искажением), можно легко их отредактировать.

Каждый тип проекции пытается минимизировать один тип искажения за счёт остальных.

Каждая из проекций имеет свои преимущества и недостатки [19], которые представлены в таблице 1.

Таблица 1 – Типы проекций изображений, достоинства и недостатки

Тип	Развертка	Достоинства	Недостатки
Прямоугольная		Отображает прямые линии в трёхмерном пространстве в прямые линии на плоской двумерной сетке	Непригодны для изображений с большим вертикальным углом зрения
Цилиндрическая		Сохраняют более точные относительные размеры объектов	Непригодны для изображений с большим вертикальным углом зрения
Прямойлинейная		Отображает прямые линии в трёхмерном пространстве в прямые линии на плоской двумерной сетке	Существенно преувеличивают перспективу по мере увеличения угла обзора, что приводит к видимому завалу объектов к границам кадра
Меркатора		Близко соотносится с цилиндрической и эквидистантной проекциями; обеспечивает меньшее растяжение по вертикали и более широкий угол зрения, чем цилиндрическая проекция	Более сильное искривление линий по горизонтали

Существуют и другие проекции, например, синусоидальная и стенографическая, но при создании панорамы они используются очень редко.

2.6 Получение бесшовной панорамы

Зачастую при сшивании изображений в панорамы на итоговом изображении можно заметить эффект «мозаики», полученный в результате изменения освещенности и оттенков из-за разного положения камеры. Поэтому для лучшего восприятия панорамы прибегают к предварительной коррекции цветовых характеристик исходных изображений и методам коррекции видимого «шва» между изображениями.

2.6.1 Коррекция цвета и яркости

Существуют несколько алгоритмов цветокоррекции изображений. В выпускной квалификационной работе рассматривается метод, основанный на преобразовании цветовой модели RGB в пространство цветов CIELAB [20]. После того как в пространстве произошли изменения по цветовым каналам, цвета конвертируются обратно в RGB.

Цветовое пространство CIELAB определено Международной комиссией по освещению (сокращенно CIE) в 1976 году.

$L^*a^*b^*$ выражает цвет как три значения: L^* – светлота (изменяется от 0 до 100, то есть от самого темного до самого светлого), а a^* и b^* для четырех уникальных цветов человеческого зрения: красный, зеленый, синий и желтый.

Пространство CIELAB трехмерное и охватывает весь спектр восприятия человека (рис. 11).

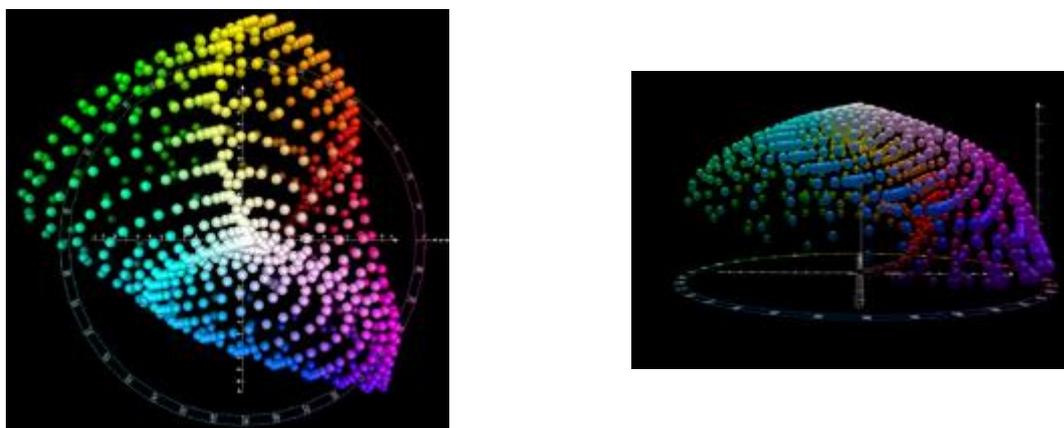


Рис. 11 – Цветовое пространство CIELAB

В случае 8-битных и 16-битных изображений R, G и B преобразуются в формат с плавающей запятой и масштабируются, чтобы соответствовать диапазону от 0 до 1. Преобразование RGB в L*a*b*:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \leftarrow \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$X \leftarrow X / X_n, \text{ где } X_n = 0.950456$$

$$Z \leftarrow Z / Z_n, \text{ где } Z_n = 1.088754$$

$$L \leftarrow \begin{cases} 116 * Y^{\frac{1}{3}} - 16 & \text{для } Y > 0.008856 \\ 903.3 * Y & \text{для } Y \leq 0.008856 \end{cases}$$

$$a \leftarrow 500(f(X) - f(Y)) + \text{delta}$$

$$b \leftarrow 200(f(Y) - f(Z)) + \text{delta}$$

где

$$f(t) = \begin{cases} t^{1/3} & \text{для } t > 0.008856 \\ 7.787t + 16/116 & \text{для } t \leq 0.008856 \end{cases}$$

и

$$\text{delta} = \begin{cases} 128 & \text{для 8 битных изображений} \\ 0 & \text{для изображений с плавающей запятой} \end{cases}$$

Диапазоны значений на выходе $0 \leq L \leq 100$, $-127 \leq a \leq 127$, $-127 \leq b \leq 127$.

Благодаря характеру определения цвета в L*a*b* появляется возможность отдельно воздействовать на яркость, контраст изображения и на его цвет. Во многих случаях это позволяет ускорить обработку изображений. L*a*b* предоставляет возможность избирательного воздействия на отдельные цвета в изображении, усиления цветового контраста, незаменимыми являются и возможности, которые это цветовое пространство предоставляет для борьбы с шумом на цифровых фотографиях.

2.6.2 Методы смешения цветов

2.6.2.1 Метод альфа-смешения

Очень простой в части идеи и реализации метод [21,22]. Базируется на понятиях прозрачности изображения (альфа) и размытии краёв (feathering). На

рисунке 12 продемонстрирован пример того, как эффект бесшовности достигается за счёт комбинации этих средств.

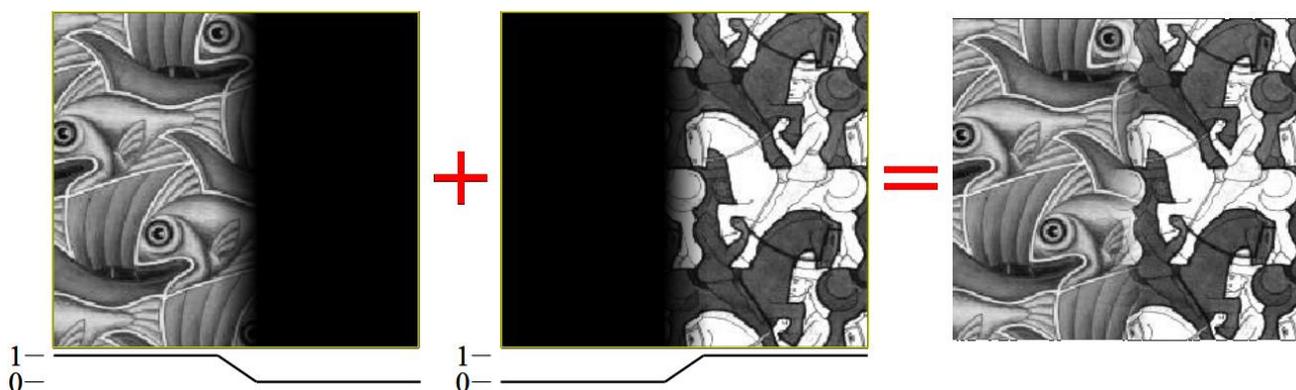


Рис. 12 – Альфа-смешение

Также подобное смешение можно представить в виде следующей формулы:

$$IB = a * IL + (1 - a) * IR \quad (1)$$

где IB – смешанное изображение, IL – левое изображение, IR – правое, a – прозрачность.

2.6.2.2 Метод многополосного смешения с использованием пирамид Гаусса и Лапласа

Этот метод эффективнее, хотя и затратнее по производительности, чем альфа-смешение. Он строится на построении пирамид Гаусса и Лапласа для изображений и последующей сборки из них итогового изображения [23].

Пирамида Гаусса – это последовательность масштабированных версий одного изображения (рис. 13). Длина этой последовательности – количество уровней, начиная от нулевого (исходное изображение). Каждый следующий уровень представляет из себя изображение, уменьшенное в два раза по своим размерам относительно предыдущего. Т.е. изображение первого уровня имеет половину разрешения от исходного, второго – четверть, третьего – восьмую часть и т.д. Сама по себе пирамида Гаусса также часто используется для создания эффекта размытия.

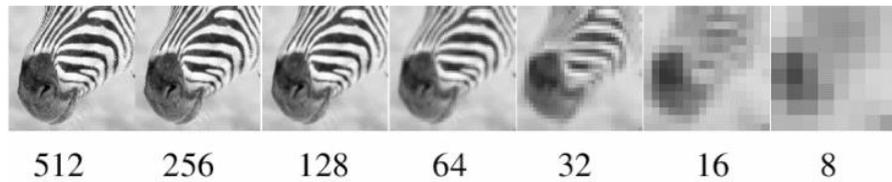


Рис. 13 – Пирамида Гаусса

Пирамида Лапласа является производной от пирамиды Гаусса и имеет схожую структуру. Однако, содержащиеся в ней изображения, по сути, обработаны высокочастотным фильтром и таким образом отражают лишь детали и грани изображения (рис. 14).

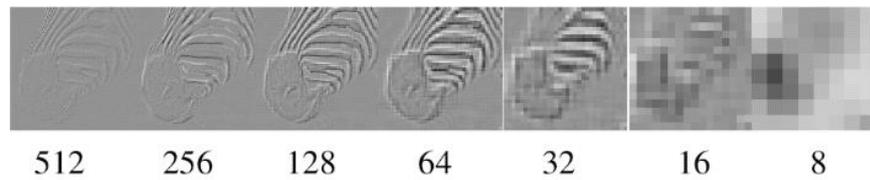


Рис. 14 – Пирамида Лапласа

Для того, чтобы смешать два изображения, для обоих строятся пирамиды Лапласа размером в N уровней, также строится пирамида Гаусса (с тем же числом уровней) для специальной маски, представляющей регион смешения (рис. 15) [24].

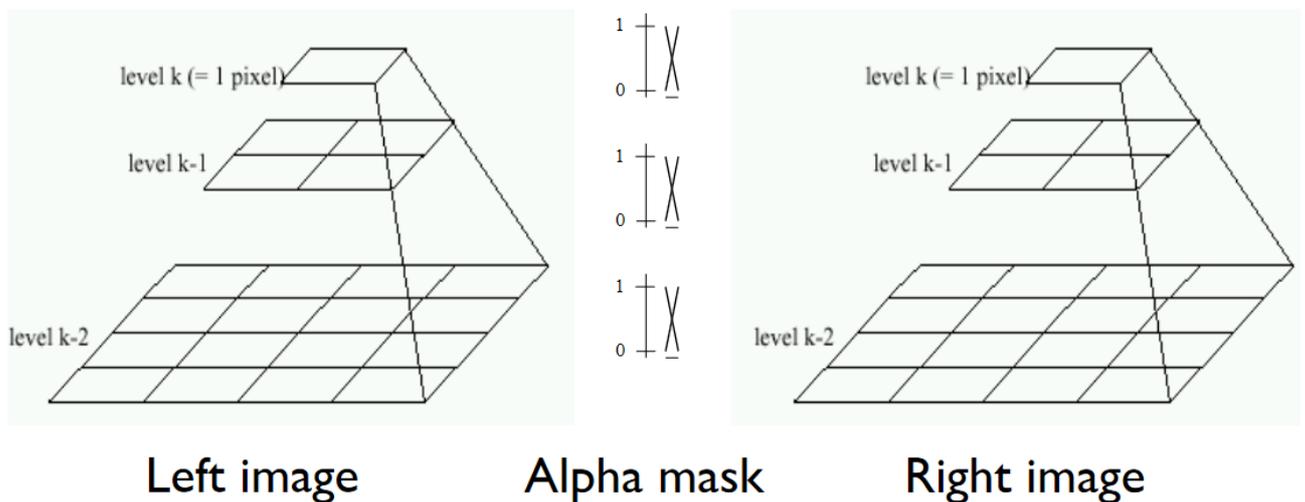


Рис. 15 – Пирамиды для смешения

Две пирамиды Лапласа (LA и LB) подвергаются смешению на каждом уровне используя соответствующие уровни из пирамиды маски (GR). Результатом будет являться смешанная пирамида, где результирующий уровень, являющийся комбинацией считается по формуле:

$$LS(i,j) = GR(I,j) * LA(I,j) + (1 - GR*(I,j)) * LB(I,j) \quad (2)$$

Благодаря возможности восстановить полноценное изображение из пирамид Лапласа и Гаусса, из получившейся пирамиды восстанавливается уже смешанное изображение, причём качество смешения тем выше, чем больше задано уровней.

3 Разработка и реализация алгоритмов генерации панорамных изображений из серии перекрывающихся кадров сцены

3.1 Функциональные и нефункциональные требования

На основе анализа области генерации панорамных изображений были сформулированы следующие функциональные требования к программному обеспечению:

- Автоматизированная работа с директориями, содержащими серии перекрывающихся изображений в произвольном количестве.
- Возможность тонкой настройки с большим количеством параметров.
- Опциональная возможность применения цилиндрического искажения к изображениям.
- Опциональная возможность применения цветовой коррекции.
- Опциональная возможность смещения изображений.
- Возможность записи итоговой панорамы в файл.
- Возможность сохранять статистические данные для дальнейшего анализа.
- Возможность трёхмерной демонстрации сшитой панорамы в цилиндрической проекции.

Нефункциональные, но важные требования следующие:

- Применение алгоритмов в режиме командной строки.
- Модульная архитектура с возможностью быстрого построения высокоуровневых алгоритмов из более низкоуровневых составляющих.

3.2 Средства реализации программной системы

Основой реализации является OpenCV – библиотека компьютерного зрения с открытым исходным кодом, в которой присутствуют необходимые возможности, такие как работа с изображениями и реализация методов поиска особых точек [25]. Она написана на языках C и C++, но для многих других языков были разработаны специальные библиотеки-обёртки для того, чтобы можно было пользоваться её возможностями.

В качестве языка разработки необходимых функций, классов и алгоритмов выбран язык программирования высокого уровня Python, который способен работать с OpenCV с помощью пакета OpenCV-Python.

В качестве среды разработки выбран редактор Visual Studio Code с возможностями IDE: специальное дополнение для языка Python, включающее автодополнение, статический анализ и подсветку синтаксиса позволяет эффективно вести разработку на данном языке программирования.

В частности, для создания алгоритма использовались:

- `opencv_core` – основная функциональность. Включает в себя базовые структуры и вычисления;
- `opencv_imgproc` – обработка изображений (линейная и нелинейная фильтрация, геометрические преобразования (масштабирование, аффинные и перспективные преобразования), преобразование цветовых пространств и т. д.);
- `opencv_features2d` — предназначен для поиска ключевых точек на изображениях, сопоставления дескрипторов и вывода особенностей на изображениях;
- `opencv_highgui` – ввод/вывод изображений и видео;
- `opencv_calib3d` – базовые геометрические алгоритмы стереоскопической визуализации, методы калибровки обычной и стереоскопической камеры, оценки расположения объектов, алгоритмы поиска соответствий для стереоизображений, элементы 3D реконструкции.

В данной библиотеке присутствуют различные методы компьютерного зрения и обработки изображений, в т.ч. SIFT, SURF, ORB, BRISK и AKAZE. Они

относятся к числу основных детекторов признаков, инвариантных при масштабируемости, вращении и аффинных преобразованиях, каждый из которых имеет обозначенный дескриптор признаков и обладает своими достоинствами и недостатками.

В ходе исследования было выявлено, что метод SURF деактивирован в актуальной версии библиотеки OpenCV по причине того, что он запатентован, поэтому не предоставляется возможным его использование.

После обнаружения ключевых особенностей выполняется сопоставление характеристик двумя методами сопоставления, реализуемые через OpenCV: BFMatcher и FlannBasedMatcher.

BFMatcher принимает дескриптор одного объекта в первом наборе и сопоставляется со всеми другими объектами во втором наборе с использованием некоторого расчета расстояния, затем возвращается ближайший.

FlannBasedMatcher содержит набор алгоритмов, оптимизированных для быстрого поиска ближайшего соседа в больших наборах данных и для многомерных объектов. Он работает быстрее, чем BFMatcher для больших наборов данных [26]. Другими словами, FLANN намного быстрее, чем BFMatcher, но он находит только приблизительного ближайшего соседа, что является хорошим совпадением, но не обязательно лучшим.

На рисунке 16 изображены схематично метрики, оценивающие соответствие между двумя рассматриваемыми дескрипторами ключевых точек описанными выше методами сопоставления особенностей в библиотеке OpenCV.



Рис. 16 – Метрики расчёта расстояния между ключевыми точками в методах BFMatcher и FlannBasedMatcher

BFMatcher использует функции дистанции L1-нормы (Манхэттенское расстояние) или L2-нормы (расстояние Евклида) и расстояния Хэмминга.

FlannBasedMatcher использует алгоритм, который применяет приоритетный поиск к иерархическим деревьям k-средних и алгоритм локального хэширования (LSH) для выполнения поиска на основе FLANN.

RANSAC, LMEDS и PROSAC, ранее описанные в выпускной квалификационной работе, доступны в библиотеке OpenCV.

Преобразование перспективы вычисляется по матрице гомографии и аффинным преобразованиям.

Таким образом, изучив все алгоритмы и методы для создания панорамных изображений было решено реализовать систему из нескольких алгоритмов:

- алгоритм для совмещения изображений и получения статистических данных;
- алгоритм поэтапной сшивки пар изображений с определением матрицы гомографии и аффинными преобразованиями с реализацией функций цилиндрической проекции, цветокоррекции и смещения цветов, функций выравнивания и смещения краёв панорамы;
- алгоритм для сортировки входных изображений.

3.3 Общая программная система алгоритмов для создания панорамного изображения

Регистрация изображений – это процесс сопоставления, совмещения и наложения двух или более изображений сцены, снятых с разных точек обзора. Он широко используется во многих приложениях для машинного зрения. Регистрация изображения состоит из пяти основных этапов: обнаружение и извлечение особенностей, поиск соответствий признаков, определение «хороших» точек и неверных совпадений, трансформация изображения, реконструкция и сшивка изображения.

Время и точность совмещения изображений на основе признаков в основном зависят от вычислительной эффективности и надежности выбранного

метода нахождения признаков, поэтому первоначальной задачей работы является разработка алгоритма для поэтапного совмещения перекрывающихся изображений с различными метриками выбранных методов.

3.3.1 Разработка и реализация алгоритма для совмещения изображений и получения статистических данных

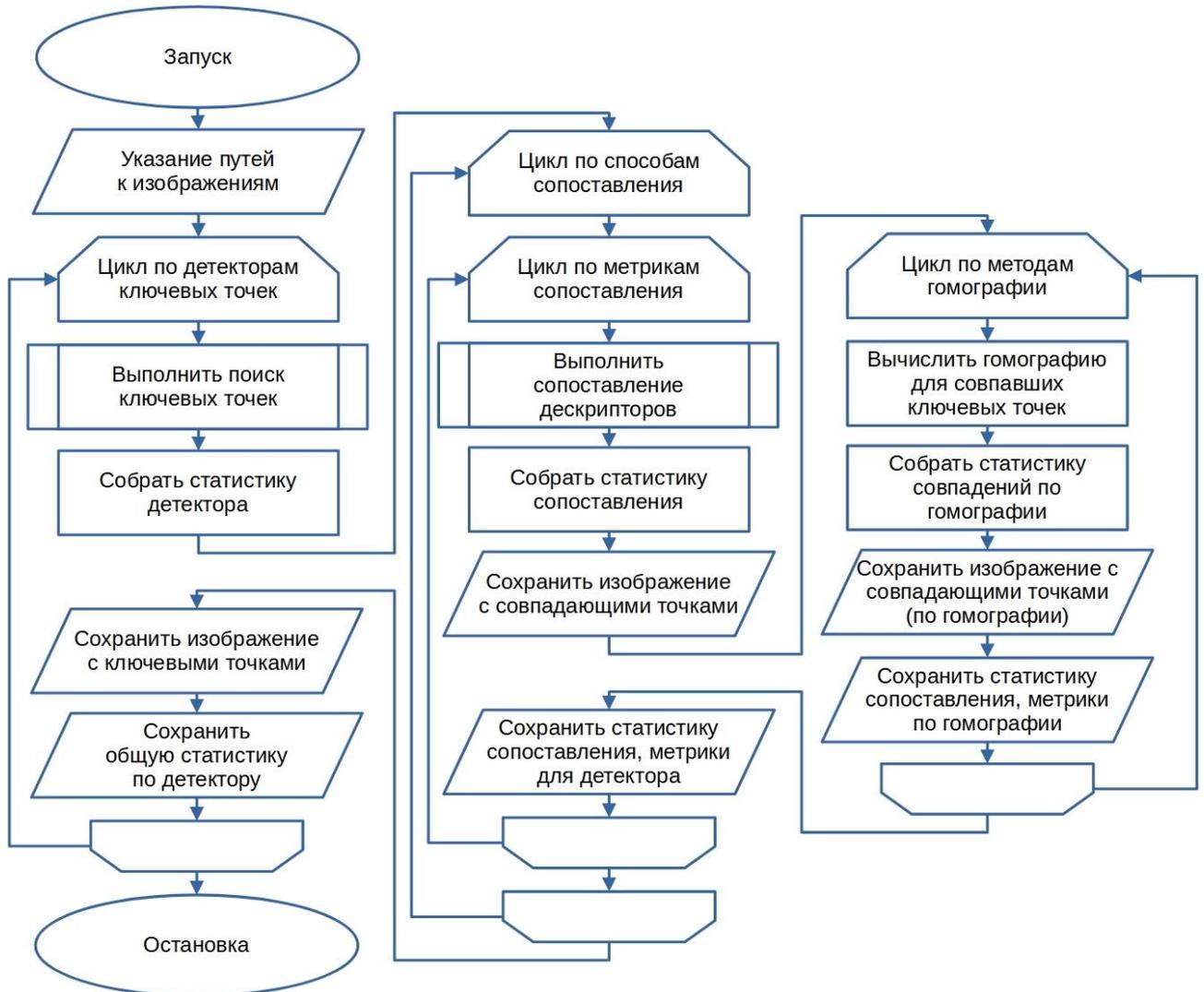


Рис. 17 – Блок-схема алгоритма совмещения изображений и получения статистических данных

Итоговый алгоритм совмещения изображений и получения статистических данных представлен на рисунке 17 и состоит из следующих этапов:

1. Указание путей к изображениям. Создаётся директория для результатов теста, с названием в виде текущих даты и времени.

2. Цикл по детекторам ключевых точек. По всем методам поиска ключевых точек – AKAZE, BRISK, SIFT, ORB. В начале тела цикла считываются файлы проверяемых изображений, создаётся поддиректория по названию метода, в которую будут сохранены все результаты, а затем создаётся объект детектора, с помощью которого в прочитанных изображениях ищутся ключевые точки и извлекаются дескрипторы. Для общей статистики по методу считается затраченное на вычисление время и количество точек.

3. Цикл по способам сопоставления. Цикл проходит по возможным методам сопоставления ключевых точек – BFMatcher, FLANN (двух видов). Сразу же в начале его тела реализован третий цикл – по метрикам, таким как NORM_L1, NORM_L2, NORM_HAMMING. В случае с FLANN метрики не задействуются, поэтому у третьего цикла будет только одна итерация.

Будет выполнено создание очередной поддиректории (уже в рамках метода поиска ключевых точек) сравнение дескрипторов ключевых точек обоих изображений с текущей комбинацией метода сравнения и метрики (время данной операции также измеряется). Для статистики также берётся количество подходящих («хороших») совпадений, а в директорию сохраняется комбинированное изображение с сопоставленными «хорошими» точками.

4. Цикл по методам гомографии. Реализовано также с помощью циклов несколько методов вычисления гомографии. Эта операция позволяет дополнительно отфильтровать неподходящие точки. Сохраняется новая статистика для каждого метода вычисления гомографии (количество и процент точек, подтверждённых гомографией) и новое комбинированное изображение с данными точками. Именно это и позволит сравнивать относительную эффективность различных методов.

Для каждого метода вычисления гомографии сохраняется новая статистика (количество и процент точек, подтверждённых гомографией) и новое комбинированное изображение с данными точками.

3.3.2 Разработка и реализация алгоритма сшивки пары изображений

Этот низкоуровневый алгоритм, реализованный в виде специальной функции и набора небольших классов и методов в отдельном Python-модуле «panorama.py», предназначен для использования в высокоуровневых алгоритмах сшивки.

Небольшой класс `StitchParams` не имеет методов, а только статические поля (т. е. не подразумевается создания экземпляров данного класса). Он предназначен для изменения настроек сшивки в рамках высокоуровневых алгоритмов путём простого присвоения другим значений полям.

Поля `RATIO` и `MINDIST` предназначены для сопоставления наборов ключевых точек, являясь критериями близости сравниваемых дескрипторов. `DETECT_OPTIMIZE` и `OPTIMIZE_BY_LAST` отвечают за возможность несколько ускорить сшивку больших панорам путём поиска ключевых точек только на части изображения (первый параметр – что берётся половина изображения, второй – что берётся по длине последнего пришитого). `HOMOGRAPHY_FUNC` – выбранная функция соответствия (`RANSAC`, `LMEDS`, `PROSAC`). `MODE` – выбранный способ преобразования, аффинные либо гомография. `SAVE_PER_STEP` – флаг для сохранения промежуточных изображений на каждом шаге (предназначен уже непосредственно для алгоритма построения панорамы). `CYL_WARP` и `CYL_F` – параметры цилиндрической проекции, первый – использовать ли её, второй – радиус виртуального цилиндра. `CORR_COL` и `CORR_CHAIN` – выбранный метод цветокоррекции (либо его отсутствие) и флаг для запоминания цветов текущего обработанного изображения (чтобы каждое последующее корректировалось по предыдущему скорректированному).

Непосредственно функция сшивки помимо учёта вышеописанных параметров принимает несколько своих – пару изображение для сшивки, объекты детектора особенностей, сопоставления особенностей и смещения и также опциональную ширину последнего ранее пришитого изображения (для оптимизации при сшивке масштабных панорам).

На следующем этапе производится расширение области изображения (для избежания в будущем потери участков), оптимизация области, в которой будут искажаться ключевые точки на одном из изображений (если такая настройка задана) и непосредственно поиск ключевых точек и извлечение дескрипторов.

Далее извлечённые особенности сопоставляются, фильтруются «хорошие» совпадения. И проводится оценка гомографии (либо аффинных преобразований), и применение данной трансформации к сшиваемому изображению.

Заключительный этап – сохранение размеров трансформированного изображения (для возможной оптимизации) и затем смешение изображений с помощью переданного объекта, который может как быть настроен на определённый способ, так и просто грубо наложить изображения друг на друга при необходимости. Итоговое изображение является выходным значением функции.

Кроме того, с расчётом на сшивку панорам от центра с движением налево и направо, сделана функция для противоположного направления движения от центра, которая инвертирует изображения необходимым образом, вызывает вышеописанную функцию, и снова инвертирует полученный от неё результат.

3.3.3 Разработка и реализация функций цилиндрической проекции

Функции, которые позволяют применить цилиндрическое преобразование к переданному изображению, необходимы для сшивки рассчитанных на цилиндрическую проекцию панорам. Реализована в рамках модуля «panorama.py».

Первая функция `cylindricalWarp()` в качестве параметра принимает изображение и матрицу для его преобразования. Используя матрицу, функция преобразует изображение и возвращает полученный результат.

Вторая функция `cilWarpAuto()` является «обёрткой» первой и предназначена для формирования матрицы исходя из размеров изображения и заданного в настройках сшивки радиуса цилиндра. Затем она вызывает первую и возвращает полученное трансформированное изображение.

3.3.4 Разработка и реализация функций цветовой коррекции

Для реализации был выбран метод переноса цвета между изображениями, и она была проведена в модуле «color_corrections.py».

Первая функция `image_stats()` в качестве параметра принимает изображение и возвращает параметры цветов в рамках цветового пространства LAB, а именно средние и стандартные отклонения для каждого канала. Она предназначена для использования в составе основной функции, которая будет рассмотрена последней в данной главе.

Следующие две функции выполняют сугубо служебную роль – обработку входящих изображений, чтобы значения каждого пикселя не выбивались из диапазона 0-255, либо путём масштабирования, либо просто перезаписью выбивающихся значений на минимально или максимально допустимые.

Функция `_min_max_scale()` предназначена для масштабирования по минимальным и максимальным значениям в рамках массива, представляющего изображение.

Функция `_scale_array()` либо вызывает предыдущую в зависимости от переданного параметра-флага, либо просто ограничивает допустимые значения методом `clip()` библиотеки Numpy.

Основная функция задействует вышеописанные. Она принимает пару изображений – то, с которого брать цвета (источник), и то, на которое их переносить (цель). Параметр `clip` отвечает за то, использовать ли масштабирование или просто ограничить значения. Параметр `preserve_paper` – за подход, которым вычисляются средние значения цветов. Вначале подготавливается маска целевого изображения, чтобы её применить в конце функции – это необходимо, чтобы в случае передачи изображения с пустыми областями мы эти области могли восстановить на обработанном изображении. Далее в функции `color_transfer()` оба изображения конвертируются из RGB в LAB, а затем используется `image_stats` для получения параметров их цветов.

Далее целевое изображение разбивается на массивы каналов методом `split()` библиотеки OpenCV. Из каждого канала вычитается среднее отклонение

для этого канала, полученное ранее. В зависимости от параметра `preserve_paper`, массивы каналов преобразуются.

К изменённым каналам добавляется среднее отклонение изображения-источника, а затем выполняется масштабирование (либо ограничение) итоговых массивов.

Финальный шаг – соединение получившихся каналов в итоговое изображение и применение к нему маски, после этого происходит возврат полученного результата.

3.3.5 Разработка и реализация функций смешения изображений

Для реализации были выбраны методы альфа-смешения (Alpha Blending, также Feathering) и смешение с помощью пирамид Гаусса/Лапласа (Pyramid Blending, также Multi-band Blending), для них был создан модуль «`blendings.py`».

Кроме того, реализована простая функция `blend_primitive()` для наложения изображений друг на друга без смешения.

Функция альфа-смешения `feather_stitching_api()` использует встроенную возможность OpenCV, для которого нужно лишь подготовить маски входных изображений и последовательно передать (сначала исходные изображения, затем маски) в объект `blender`, созданный методом `detail_FeatherBlender()`. Путём вызова метода `blend()` будет получено смешанное изображение.

Многополосное смешение (с помощью пирамид Гаусса/Лапласа) разбито уже на несколько функций, по аналогии с цветокоррекцией. Первая из служебных функций `preprocess()` предназначена для первичной обработки изображений, которые требуется смешать. На вход принимает оба изображения, ширину области их пересечения (в рамках которой и будет проводиться смешение) и флаг для использования только половины изображений (т. к. в ряде случаев этого может быть достаточно). Проводятся проверки на равенство их размеров и на то, чтобы ширина области пересечения не превосходила ширину.

Далее изображения преобразуются – за счёт пустых значений ширина изменяется для оптимального наложения изображений друг на друга, также создаётся специальная маска. При этом именно здесь предусмотрен вариант, что

берутся только половины изображений. В конце функция возвращает изображения и маску.

Далее реализована функция `GaussianPyramid()` для построения списка изображений, являющийся пирамидой Гаусса. Она принимает `numpy`-массив, представляющий изображение и требуемое число уровней, а на выходе – список модифицированных изображений, от большего к меньшему.

Следующая функция `LaplacianPyramid()` работает схожим образом (и с теми же входными параметрами), но уже генерирует пирамиду Лапласа.

Далее задаётся функция `blend_pyramid()` для смешения пирамид, принимающая в качестве параметров две пирамиды Лапласа (одна на каждое смешиваемое изображение) и пирамиду Гаусса.

Очередная функция `reconstruct()` предназначена для восстановления уже смешанного изображения из списка, полученного в предыдущей функции.

Так как, в результате смешения пустые области исходных изображений станут уже не пустыми, то необходима особая функция `getBlendMask()`, создающая маску, по которой области вне площади изображений будут очищены.

Итоговая функция `multiband()` использует вышеописанные. Она принимает в качестве параметров изображения, ширину области пересечения, число уровней и флаг для использования половины площади. Первыми шагами проверяется корректность ширины, проводится первичная обработка функцией `preprocess()`, подготавливается маска, считается максимально допустимое число для проверки допустимости переданного в функцию.

Затем создаются пирамида Гаусса из маски для смешения и пирамиды Лапласа для изображений, эти пирамиды смешиваются и затем реконструируются в изображение. К получившемуся изображению применяется маска, результат выдаётся функцией.

Далее реализуется специальный класс `BlenderObj()`, чтобы можно было быстро использовать любые из способов смешения в алгоритмах. В конструктор он принимает параметры выбора способа смешения и числа уровней, а в методе

`blend()` выполняет смещение выбранным способом. Для многополосного смещения присутствует автоматическое вычисление ширины области пересечения.

3.3.6 Разработка и реализация функций выравнивания и смещения краёв панорамы

После того, как панорама будет сшита, то для того, чтобы она была пригодной к применению в цилиндрической проекции, необходимо произвести смещение её краёв. С этой целью создан модуль «`align_rotate_ends_blend.py`».

Перед непосредственным смещением краёв может понадобится немного выровнять панораму в случае, если съёмка была не вполне равномерной. Специальная функция `align_rotate()` производит циклический поворот вокруг центра переданного изображения на небольшой градус и проверяет высоту получившегося изображения.

Если продуктивного вращения не произошло, то предпринимается аналогичная попытка вращения в противоположном направлении.

Если в итоге найден оптимальный градус поворота, то возвращается панорама повернутая на этот угол, а иначе – исходное изображение, таким образом панорама оказывается подготовленной к смещению краёв.

Функция `blend_ends()` смещения краёв является модифицированной версией функции для сшивки двух изображений. Является упрощённой реализацией метода, предложенного в статье «Edge blending using commodity projectors» [27]. Переданная панорама разделяется на центральную, правую и левую части. Для правой и левых частей производится поиск и сопоставление ключевых точек, выбор наилучших совпадений.

Следующий этап – вычисление аффинных преобразований, из которых берётся нужное смещение по горизонтали одного изображения относительно другого. Затем области левого и правого изображения трансформируются по ширине и производится их смещение.

Финальный шаг – разделение смешанного изображения пополам и совмещение половин с оставшейся центральной частью исходного изображения. Таким образом, функция возвращает панораму со смешанными краями.

3.3.7 Разработка и реализация алгоритма сортировки изображений

Этот алгоритм не является встроенным в сшивку панорамных изображений из соображений производительности и целесообразности, но будучи используемым отдельно он позволит подготовить некоторый набор изображений к сшивке. Это может быть актуально, если исходные изображения не отсортированы. Алгоритм реализован отдельным скриптом, задействующий некоторые из описанных ранее модули.

Первая функция `getNumberOfMatches()` позволит подсчитать число совпадений по ключевым точкам для пары изображений.

Вторая `getLeftmost()` позволит найти самое левое изображение в списке, по минимуму совпадений ключевых точек с прочими изображениями списка.

Далее эта функция используется в цикле по числу изображений, на каждом шаге ищется самое левое из оставшихся и сохраняется под следующим индексом. В итоге в определённой директории будет отсортированный пронумерованный набор изображений.

3.3.8 Разработка и реализация алгоритма сшивки панорамного изображения

Алгоритм предназначен для сшивки последовательной серии перекрывающихся (заранее отсортированных) изображений по ключевым точкам с использованием различных параметров. Он задействует почти все описанные ранее модули, кроме «`align_rotate_ends_blend.py`», поскольку функции данного модуля применяются к уже сшитому изображению. Итоговый вариант представлен в виде блок-схемы на рисунке 18.

Алгоритм реализован в модуле «`panorama.py`», в виде отдельного класса. Конструктор класса принимает путь к директории с набором отсортированных

изображений, а также объекты поиска и сопоставления ключевых точек и объект смещения.

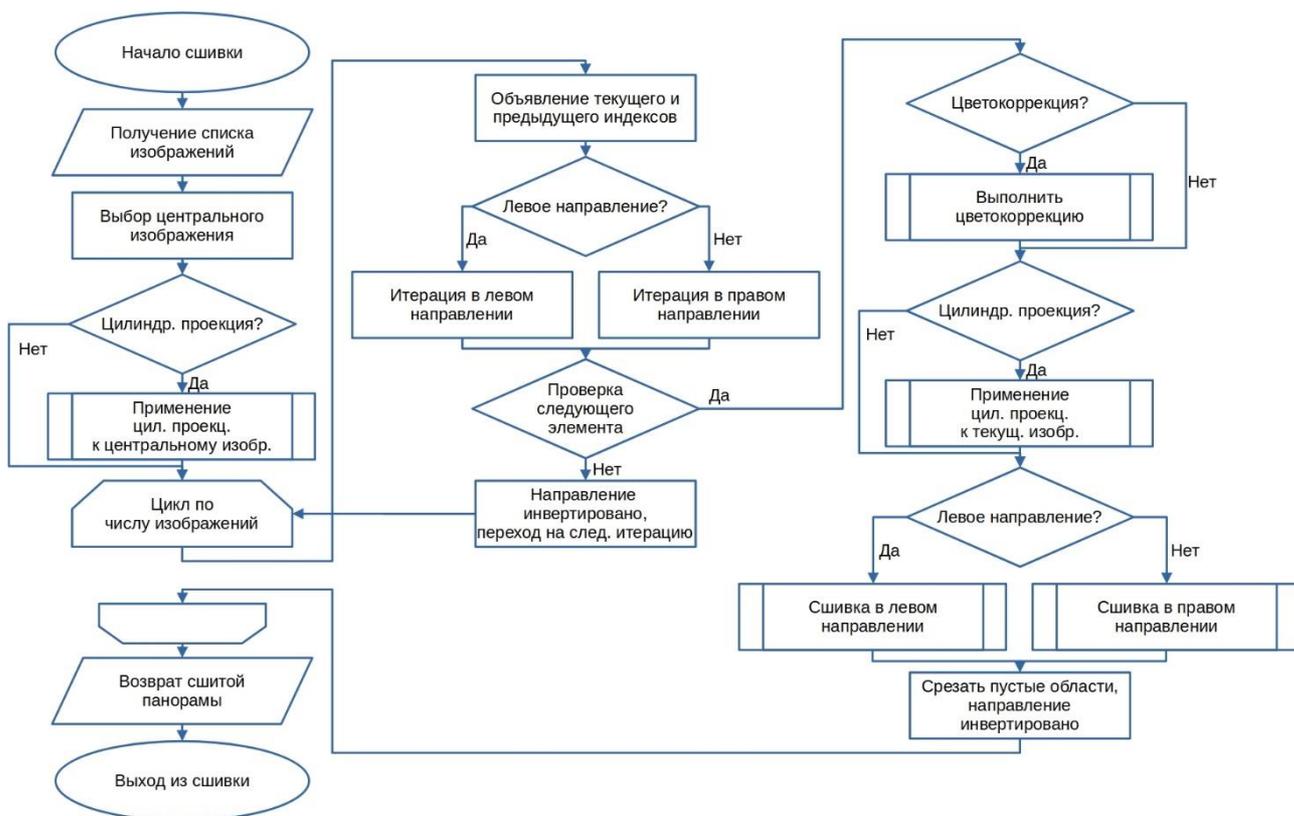


Рис. 18 – Блок-схема алгоритма сшивки панорамы

Проводится чтение изображений из директории и присвоение объектов внутренним полям.

Метод `createPanorama()` в свою очередь проводит сшивку панорамы из загруженных изображений. Первый его шаг – взять центральное (по индексу) изображение в качестве центрального в панораме, т. е. остальные будут сшиваться с ним в правом или в левом направлении. При необходимости применяется цилиндрическое искажение.

Далее инициализируется цикл по числу изображений, в его начале проводится определение индексов текущего и предыдущего изображений в зависимости от того, в каком направлении сшивка проводится в данный момент.

Затем проводится проверка на то, существует ли очередное изображение, либо же движение в эту сторону завершено. Если оно завершено, то направление меняется и начинается новая итерация цикла, если нет – то выполняются (если

настроены) цветокоррекция и цилиндрическое искажение очередного изображения.

На следующем шаге в зависимости от направления вызывается соответствующий метод сшивки двух изображений.

Далее в конце цикла со сшитого изображения срезаются пустые области, меняется направление движения и счётчик сшитых изображений увеличивается. В конце метод возвращает получившееся изображение.

3.3.9 Структура программной системы для сшивки панорамных изображений

Программная система разрабатывалась в виде набора модулей Python, каждый из которых специализирован:

- «feature_detection.py» – модуль для работы с ключевыми точками;
- «img_trim.py» – небольшой набор функций для обрезки изображений;
- «color_corrections.py» – реализация цветовой коррекции;
- «blendings.py» – модуль для смешения изображений;
- «panorama.py» – модуль для создания панорамных изображений;
- «align_rotate_ends_blend.py» – модуль для смешения краёв у созданных панорам.

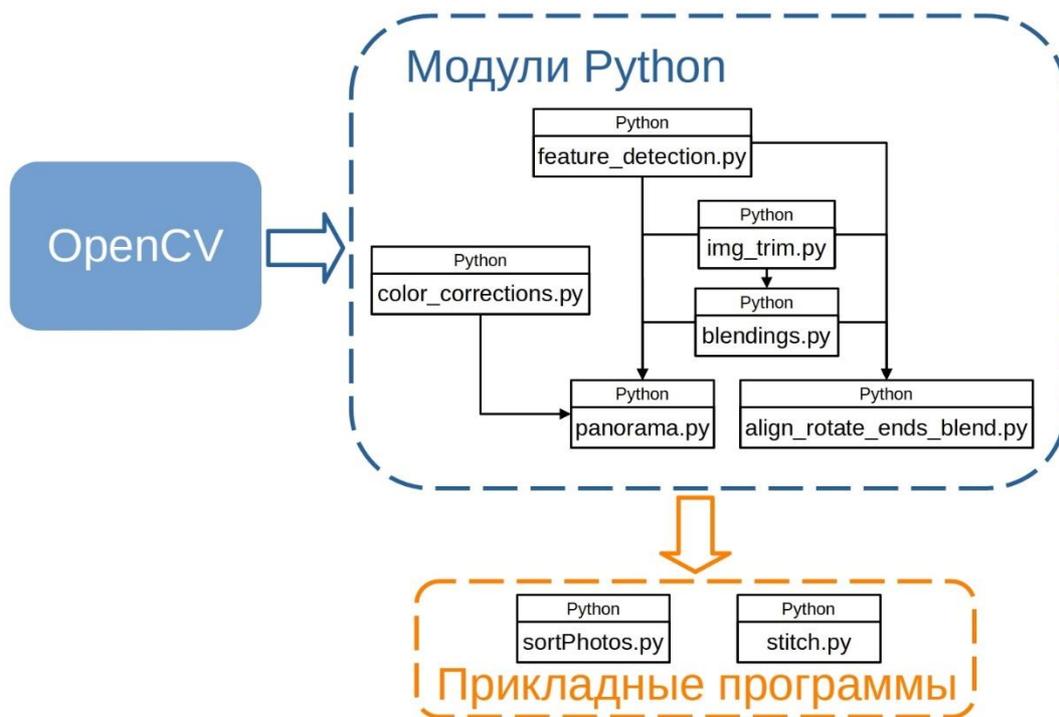


Рис. 19 – Модули и прикладные программы

В итоге, этот набор модулей представляет из себя небольшую библиотеку, базирующуюся на основе OpenCV, пригодную для использования в прикладных, ориентированных на практическое применение, программах на языке Python (рис. 19). Например, «sortPhotos.py» – это рассмотренный ранее алгоритм сортировки изображений, который задействует для этого несколько модулей, а «stitcher.py» запускает сшивку какого-либо набора изображений, а затем выравнивает полученный результат и смешивает у него края.

3.4 Разработка и реализация визуализации панорамного изображения

Для визуализации панорамы с использованием веб-технологий использована библиотека на языке JavaScript panolens.js, которая в свою очередь требует графическую библиотеку three.js. Развёртывание демонстрационной визуализации проведено в рамках репозитория на сервисе GitHub с использованием опции Pages, которая позволяет репозиторию быть хостингом статических веб-страниц.

При этом, вследствие особенностей работы panolens.js и для лучших демонстрационных возможностей, было принято решение сделать две html-

страницы – одна для непосредственной отрисовки панорамы в соответствии с переданными GET-параметрами, а вторая – для подключения первой через html-элемент `iframe`. Это позволяет при необходимости вывести несколько панорам в рамках одной страницы без необходимости писать сложный JS-код. Схема такого подхода представлена на рисунке 20.

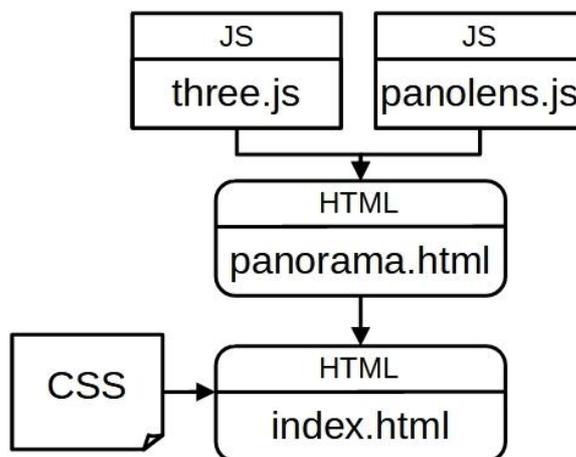


Рис. 20 – Схема веб-визуализации панорамы

В библиотеку `panolens.js` были внесены доработки для существующей функции-конструктора. Вместо используемой по умолчанию сферической проекции сделана цилиндрическая и добавлены входные параметры для радиуса и высоты цилиндра в пикселях. Итоговая функция `ImagePanorama()`, используя `three.js`, инициализирует цилиндрический «каркас» для панорамы с заданным радиусом и высотой цилиндра.

Файл `panorama.html` включает в себя JS-код для обработки входящих параметров и инициализации панорамы. Для получения параметров берётся строка запроса, и из неё извлекаются необходимые параметры, такие как радиус и высота цилиндра, целевое панорамное изображение и угол обзора.

Далее инициализируется новый объект панорамы с указанием изображения, радиуса и высоты.

Следующий шаг – задание объекта визуализации с набором настроек отображения панорамы, таких как угол обзора, отображение элементов управления и т. д., а затем и передача объекта панорамы в объект визуализации.

В файле `index.html` подключается страница `panorama.html` через элемент `iframe`, используя простую url-ссылку с параметрами.

4 Результаты экспериментов

4.1 Сравнительный анализ полученных результатов эффективности методов поиска ключевых точек

Для оценки эффективности обработки двух изображений и сопоставления методами поиска ключевых особенностей SIFT, ORB, BRISK и AKAZE было проведено тестирование на различных входных данных, а именно изображения «MOUNTAIN», «RIVER», «HOUSE», «UTTOWER» и «ROOFS» при изменении угла обзора и яркости кадра во время получения снимка относительно исходного. Тестирование произведено с помощью разработанного алгоритма поэтапного совмещения перекрывающихся изображений, который позволяет собрать необходимую статистику и проиллюстрировать её.

Результаты тестирования частично представлены на изображениях MOUNTAIN на рисунке 21 для метода ORB с выбранными методами сопоставления BFMatcher и FlannBasedMatcher с оценкой матрицы гомографии по методу RANSAC. Полный перечень результатов по всем метрикам продемонстрирован в приложении А.

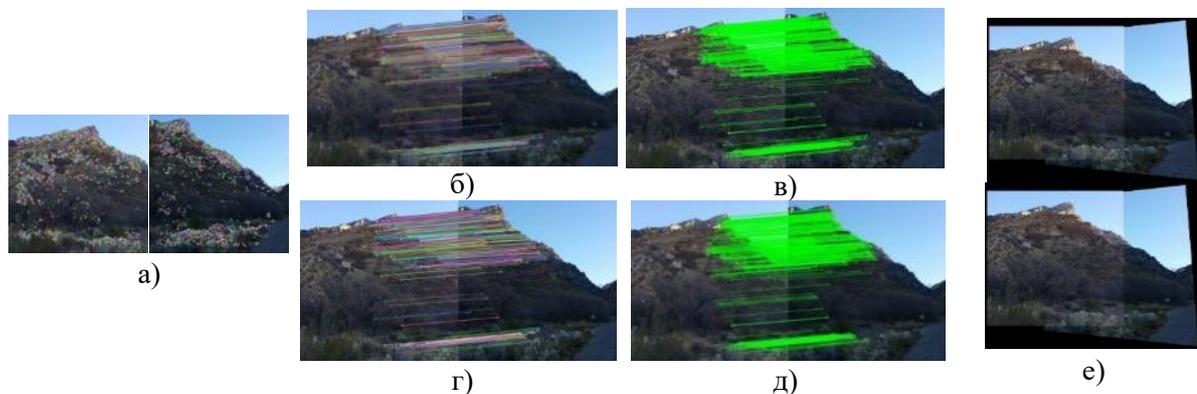


Рис. 21 - Результаты тестирования методом ORB: а) ключевые точки на кадрах MOUNTAIN; сопоставление всех точек б) BFMatcher_HAMMING и г) FlannBasedMatcher_FLANN_LSH; сопоставление RANSAC в) BFMatcher и д) FlannBasedMatcher; сшитые изображения е) гомография и аффинные преобразования

Проведение визуального сравнительного анализа эффективности обработки методами по итоговому изображению достаточно затруднительно.

Для выполнения оценки эффективности методов выполнено сравнение результатов по средним значениям в виде диаграмм (рис. 22-24).

Можно заключить, что метод поиска особенностей BRISK обладает самой высокой производительностью – больше всего найденных ключевых точек на тестовых изображениях и наименьшее среднее время поиска.

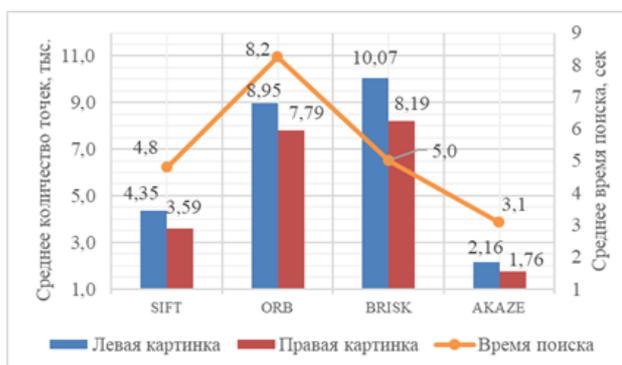


Рис. 22 – Результаты поиска ключевых точек

Лучший способ сопоставления – VFMatcher с функцией расстояния Евклида и расстояния Хэмминга, однако, стоит провести тестирование метрики FlannBasedMatcher с алгоритмом k-средних и алгоритмом локального хэширования на сшивке изображений в панорамные снимки.

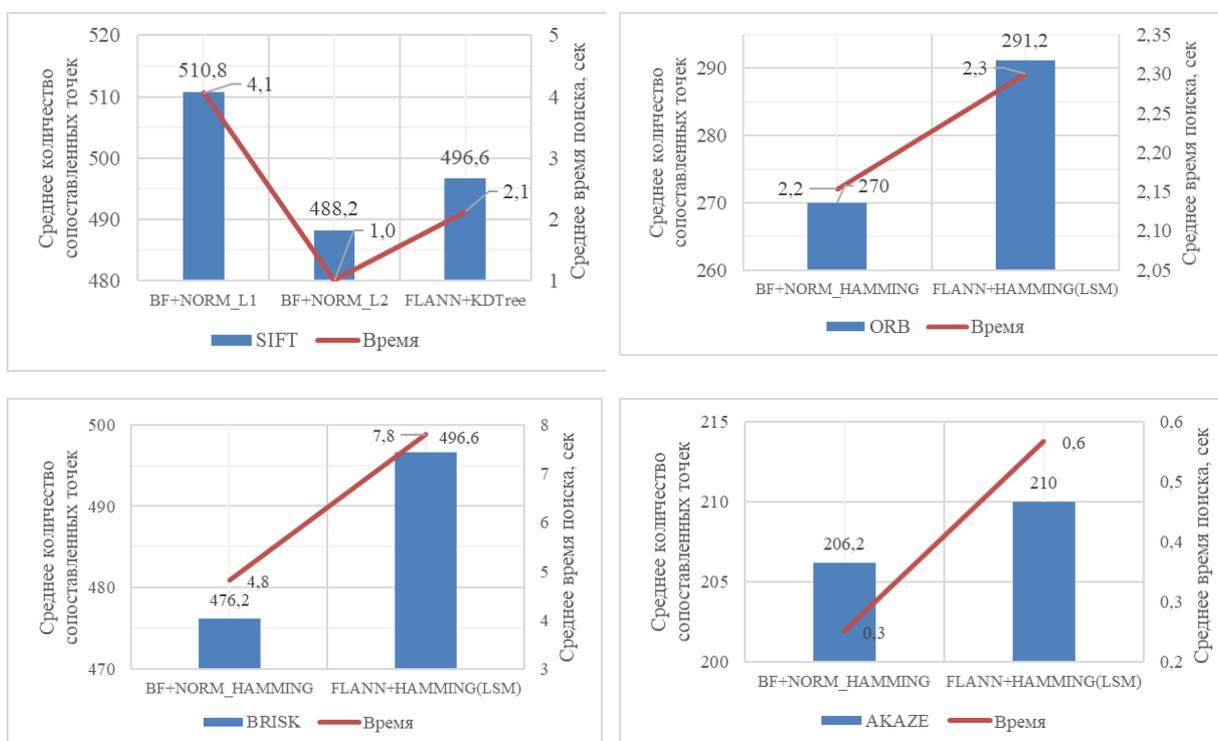


Рис. 23 – Результаты сопоставления точек методами VFMatcher и FlannBasedMatcher

Также, следует отметить, что на результаты исследований повлияло качество изображений: больше «хороших» совпадений было на изображениях с однородным и ярким фоном, большей детализацией и незначительным изменением перспективы.

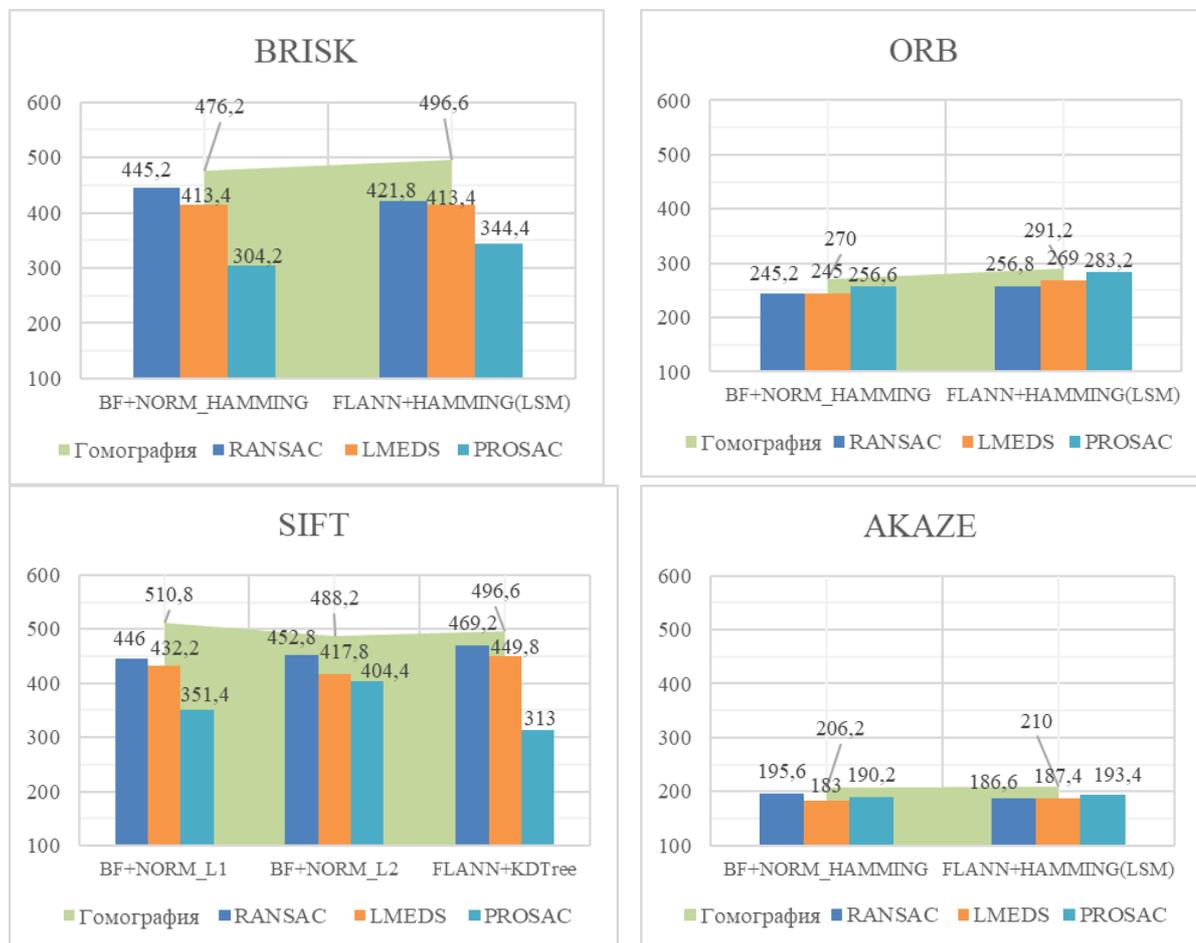


Рис. 24 – Результаты по методам удаления неверных совпадений ключевых точек

Анализ полученных данных позволяет говорить о том, что метод улучшения гомографии RANSAC в качестве поиска и удаления неверных совпадений превосходит другие алгоритмы на 3-10 %.

В результате работы алгоритма поэтапной сшивки изображений MOUNTAIN, RIVER, HOUSE, UTOWER и ROOFS с определением матрицы гомографии и аффинными преобразованиями получилось 198 наблюдений. Сшить изображение не удалось методом SIFT с использованием функции дистанции L1-нормы (Манхэттенское расстояние). Ранее была доказана высокая производительность метода BRISK, лучший способ сопоставления для него -

BFMatcher с функцией расстояния Хэмминга и вероятностным методом RANSAC. Рассмотрим подробнее изображения с данными метриками (рис. 25).

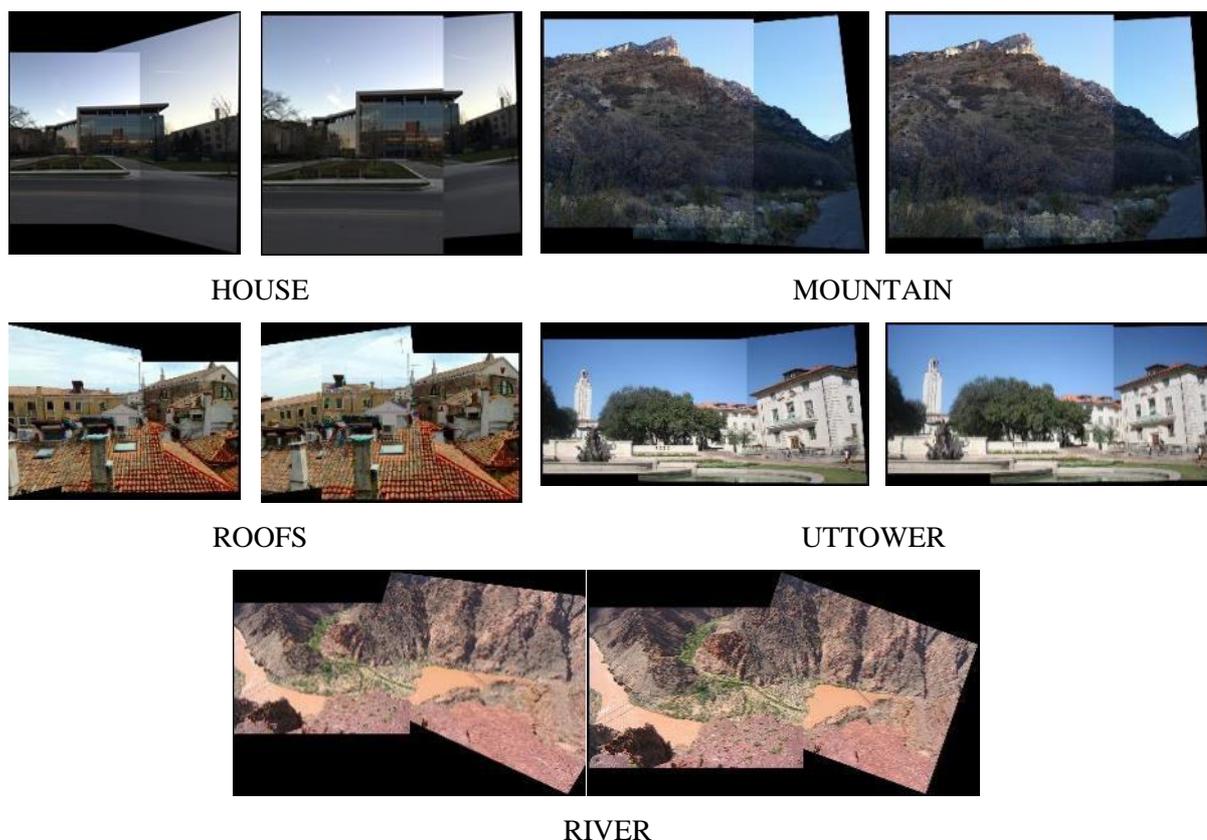


Рис. 25 – Результаты шивки изображений с определением матрицы гомографии (левый рисунок) и аффинными преобразованиями (правый рисунок)

Можно с уверенностью сказать, что матрица гомографии эффективнее для шивки кадров в панорамные изображения, так как преобразования изображений выполняются в трехмерном пространстве.

Эти параметры принимаются в качестве параметров по умолчанию в дальнейшей работе, т. е. для проверки возможностей шивки с использованием возможностей смещения, цветокоррекции, цилиндрических искажений и в панорамном режиме с задействованием нескольких изображений.

Тестирование со сбором статистики проводилось на персональном компьютере с конфигурацией, приведённой в таблице 2.

Таблица 2 – Описание конфигурации компьютера

Монитор	LG 23MP68VQ, 1920x1080, 16:9
Процессор	AMD Ryzen 7 1700X (восемь ядер, шестнадцать потоков, 3,4/3,8 ГГц, 16 Мбайт L3)
Оперативная память	32 Гбайт DDR4-3066
Чипсет	AMD X570
Видеокарта	AMD Radeon 580X, 8 Гб GDDR5
Жёсткий диск	Seagate Barracuda 1 Тбайт (7200 об/мин, Serial ATA)
Твердотельный жёсткий диск	ADATA 256 Гбайт
Блок питания	700 Вт
Размеры системного блока	336x424x320 мм
Операционная система	Windows 10 Pro 64-bit

4.2 Сравнительная демонстрация сшивки изображений с использованием цветокоррекции и смешения

Для первичной проверки возможностей и качества сшивки с использованием цветокоррекции и смешения взяты те же небольшие наборы изображений – MOUNTAIN, RIVER, HOUSE, UTOWER и ROOFS и использованы наиболее эффективные параметры. При этом, однако, используется полноценный алгоритм панорамной сшивки, просто с отключенной опцией цилиндрического искажения. Это позволит для демонстрации сшить полный набор MOUNTAIN, который сам по себе включает 3 изображения, однако из-за особенностей алгоритма (направление сшивки, определение центра) наборы из пар картинок могут выглядеть немного иначе, чем в предыдущей главе. Таким образом, для каждого набора изображений проведено три сшивки: без цветокоррекции и смешения, с цветокоррекцией и альфа-смешением, с цветокоррекцией и многополосным смешением (методом пирамид Гаусса/Лапласа). Полученные результаты, на примере набора изображений MOUNTAIN представлены на рисунке 26. Результаты для других наборов представлены в приложении Б.



Рис. 26 – Результаты сшивки MOUNTAIN; левый рисунок – без использования цветокоррекции и смещения, центральный – с цветокоррекцией и альфа-смещением, правый – с цветокоррекцией и многополосным смещением

Можно отметить, что цветокоррекция методом переноса цветов и смещение в целом положительно отражаются на сшитом изображении. Имеются и некоторые особенности, и ограничения.

Цветокоррекция явным образом влияет на извлечённые особенности изображения, поскольку применяется до непосредственной сшивки, таким образом вычисленная гомография (либо аффинные координаты) и трансформация будут несколько иными.

Многополосное смещение обеспечивает более плавный переход в сравнении с методом альфа-смещения, кроме того отсутствует эффект параллакса, однако те области изображений, которые находясь в области перекрытия не имеют перекрытия как такового с областями другого изображения будут потеряны. Тем не менее, в случае ровной фотосъёмки и хорошего горизонтального перекрытия изображений этот метод явно предпочтительнее и для сшивки больших панорам будет использоваться именно он.

4.3 Сравнение с другими программными системами для сшивки панорамных изображений

Для сравнения с разработанной в рамках ВКР программной системой были выбраны два уже существующих аналога, способных к бесшовной сшивке панорам. При сравнении с ними, разработанная система будет именоваться БСИ (аббревиатура от «библиотека сшивки изображений») для краткости.

Первый аналог – High level stitching API [28], (далее для краткости Stitching API) высокоуровневый программный интерфейс приложения, существующий в рамках OpenCV и предназначенный для сшивки панорам на основе методов поиска ключевых точек. Он также имеет реализацию для языка Python и возможность выбора различных параметров, поэтому является самым простым для организации сравнения. По возможности будут подбираться те же параметры, что и для БСИ.

Второй – Hugin [29], приложение с графическим интерфейсом пользователя, реализующее возможности набора утилит и библиотек Panorama Tools [30] с открытым исходным кодом. Данное приложение и связанные библиотеки, в отличие от OpenCV предназначены исключительно для сшивки панорам. В отличие от Stitching API и БСИ является более целостной и пригодной для немедленного практического использования программной системой, однако менее оптимальной для исследовательской деятельности.

Сравнение будет разбито на несколько частей, будут рассматриваться возможности по настройке параметров сшивки, затем будут представлены практические результаты на реальных наборах фотографий.

Первая часть – про работу с ключевыми особенностями: возможности выбора метода поиска ключевых точек (и представленные варианты), метода сопоставления, функции расстояния, показателей сходства ключевых особенностей (табл. 3).

Очевидно, что БСИ предлагает намного более низкоуровневый доступ к параметрам, чем даже Stitching API, в то время как Hugin таких возможностей не предлагает вовсе. Это потенциальная возможность для улучшений и

оптимизаций, а также дальнейшей исследовательской работы над алгоритмами сшивки панорам, поскольку нет необходимости изменения исходного кода библиотеки OpenCV.

Таблица 3 – Сравнение возможностей работы с ключевыми точками

Система	Выбор метода поиска особенностей	Выбор метода сопоставления	Выбор ф-ции расстояния	Задание показателей сходства
Stitching API	SURF (если доступен), SIFT, ORB, BRISK, AKAZE	Нет	Нет	Да
Hugin	Нет (SIFT)	Нет	Нет	Нет
БСИ	SIFT, ORB, BRISK, AKAZE	Да	Хэмминга, Евклида, К-средних и др.	Да

Очевидно, что БСИ предлагает намного более низкоуровневый доступ к параметрам, чем даже Stitching API, в то время как Hugin таких возможностей не предлагает вовсе. Это потенциальная возможность для улучшений и оптимизаций, а также дальнейшей исследовательской работы над алгоритмами сшивки панорам, поскольку нет необходимости изменения исходного кода библиотеки OpenCV.

Дальнейшее сравнение (табл. 4) – по имеющимся особенностям и возможностям настройки самой сшивки, как то выбора метода вычисления преобразований перспективы, вероятностного метода для его вычисления, возможности выравнивания, цветовой коррекции, смешения изображений и целевой проекции.

Таблица 4 – Сравнение возможностей настройки с перспективными преобразованиями

Система	Выбор преобразований перспективы	Выбор вероятностного метода	Выравнивание изображений	Цветовая коррекция	Смешение изображений	Проекция
Stitching API	Гомография, аффинные	Нет	Да	Да	Альфа-смешение, многополосное смешение	Различные
Hugin	Нет (гомография)	Нет	Да	Да	Нет (собственный алгоритм)	Различные
БСИ	Гомография, аффинные	RANSAC, PROSAC, LMEDS	Частичное	Да	Альфа-смешение, многополосное смешение	Цилиндрич.

В данном случае БСИ эквивалентна Stitching API по возможностям перспективных преобразований и смешения, а также имеется возможность выбирать вероятностный метод, но возможности выравнивания более ограниченные (только для уже сшитой панорамы). Используемый в БСИ алгоритм цветовой коррекции в некоторых ситуациях уступает конкурентам, а проекция реализована только цилиндрическая, в то время как у других проектов их много. Однако, необходимо отметить, что Stitching API и Hugin/PanoTools развиваются в течение многих лет, и соответственно возможностей там реализовано больше. С другой стороны, в БСИ необходимые доработки возможно проводить быстрее, чем в Stitching API благодаря высокоуровневой реализации на языке Python, а Hugin/PanoTools в большей степени является готовым продуктом, а не программной библиотекой для использования в других проектах.

Сравнение практических результатов сшивки можно начать с набора фотографий MOUNTAIN. Для Stitching API и БСИ был выбран метод поиска ключевых особенностей BRISK и многополосное смешение. Остальные настройки БСИ – согласно ранее полученным результатам тестов. Проекция во

всех системах были отключены. Результаты сшивки представлены на рисунке 27.



Рис. 27 – Результаты сшивок MOUNTAIN

На этом наборе изображений можно отметить, что вычисление гомографии явно настроено различно в каждой системе. По каким-то причинам, есть небольшие артефакты в центре на панораме из Hugin. А смещение у Stitching API отработало не лучшим образом – весьма заметна «смазанность», т. е. в данном случае БСИ смешало лучше даже с учётом исчезновения небольших сегментов, у которых нет перекрытия с другими изображениями. При этом, формально метод смещения у Stitching API и БСИ выбраны аналогичные (но с разной реализацией).

Следующий набор фотографий – из торгового центра. Он характерен низкой яркостью и контрастностью и большим количеством деталей. Хотя это и не круговая фотосъёмка, была выбрана цилиндрическая проекция. Результаты сшивки представлены на рисунке 28.

Здесь БСИ и Stitching API демонстрируют очень близкие по качеству результаты, в то время как Hugin оказался неспособен совместить изображения ровно.



БСИ



Hugin



Stitching
API

Рис. 28 – Результаты сшивок в торговом центре

Третий набор – это результат полноценной круговой фотосъемки, пригодный для создания цилиндрической панорамы на 360 градусов. Настройки БСИ и Stitching API остаются прежними, но для первого теперь задействуется функция смещения краёв. Полученные в результате полноценные панорамы представлены на рисунке 29.

На данном наборе лучший результат – у Hugin. Он смог скомпенсировать недостатки съемки и качественно совместить края изображения. У Stitching API явно неверно преобразованы два изображения, а у БСИ не очень аккуратно сшиты края из-за недостатка выравнивания.



БСИ



Hugin



Stitching API

Рис. 29 – Результаты сшивки полной панорамы

Из этих и прочих сшивок (полный набор сшитых панорам см. в приложении В) можно сделать некоторые выводы. Универсального решения для сшивки из рассмотренных систем – нет, поскольку у разных систем разные возможности и из одного и того же набора изображений могут получиться отличающиеся результаты. При этом, в одних случаях может отработать лучше одна система, а в прочих – уже другая.

5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

5.1 Предпроектный анализ

5.1.1 Потенциальные потребители результатов исследования

Разработка, которой посвящена данная работа, представляет собой программную систему алгоритмов для сравнительного исследования метрик, для создания поэтапной сшивки изображений, и для задачи генерации панорамных изображений.

Исходя из особенностей программной системы, можно судить о круге лиц, которые потенциально будут заинтересованы в разработке. Сферы применения разработки очень широки: от нужд обычных пользователей до специализированных задач. Целевым рынком нынешней разработки являются пользователи и компании, занимающиеся исследовательской деятельностью и разрабатывающие свои программные обеспечения для создания трехмерных моделей окружающего пространства, что может быть применено в сферах туризма, картографии, космической промышленности и пр.

Сегментировать рынок услуг по потребности использования алгоритмов в разработке можно по следующим критериям: размер компании-заказчика, категория программного обеспечения (ПО).

Результат сегментирования представлен на таблице 5.

Таблица 5 – Карта сегментирования рынка

		Категория ПО			
		Инструментальные системы	Графические редакторы	Мобильные приложения	Веб-приложения
Размер компании	Крупные				
	Средние				
	Мелкие				

	Фирма А		Фирма Б		Фирма В
--	---------	--	---------	--	---------

5.1.2 Анализ конкурентных решений

Разработанный алгоритм для сравнительного исследования метрик обнаружения ключевых особенностей на изображениях и совмещения наборов изображений с чередованием множества параметров является уникальным в своём роде, так как он позволяет автоматизировано собрать большое количество статистических данных для регистрации пар изображений. Второй алгоритм позволяет произвести сшивку изображений в панорамные снимки с использованием цветокоррекции, смещения и цилиндрической проекции.

В конечном итоге, в качестве конкурирующих решений в контексте панорамной сшивки изображений были выбраны следующие ПО:

1. Модуль «image-stitcher». Модуль «image-stitcher» является инструментом для сшивания панорамных изображений с открытым исходным кодом. Готовое к использованию решение включает в себя алгоритм, использующий гомографию и цветокоррекцию для сшивки изображений в панораму. Все компоненты лицензированы по лицензии MIT и могут быть использованы в коммерческих целях [30].
2. Stitcher API. Высокоуровневый API в библиотеке компьютерного зрения OpenCV, именуемый «Stitcher class» с открытым исходным кодом. Готовое решение, алгоритм которого включает сложную систему выравнивания изображений, много вариантов целевых проекций панорамы, некоторую цветокоррекцию и смещение. Все компоненты лицензированы по лицензии Apache и могут быть использованы в коммерческих целях [28].
3. Программа Hugin. Графический интерфейс для инструментов Panorama Tools, с открытым исходным кодом, который не требует прямого взаимодействия с программами в наборе инструментов. Программа имеет простой дружественный интерфейс и поможет реализовать большие проекты с минимальными затратами. Также есть опции для калибровки линз и создания 360°-панорам. Программа предоставляется бесплатно и не имеет ограничений по количеству создаваемых панорамных снимков, все компоненты лицензированы GNU GPL [29].

Выделим критерии:

1. Технические критерии оценки ресурсоэффективности

- a. Удобство эксплуатации
- b. Надёжность
- c. Требуемые ресурсы
- d. Функциональность
- e. Простота эксплуатации

2. Экономические критерии оценки эффективности

- a. Обслуживание
- b. Срок разработки

Экспертная оценка основных технических характеристик данных продуктов представлена в таблице 6.

Таблица 6 – Оценочная карта сравнения технических решений

№	Критерии оценки	Вес	Баллы				Конкурентоспособность			
			Бф	Бк1	Бк2	Бк3	Кф	К1	К2	К3
	1	2	3	4	5	6	7	8	9	10
Технические критерии оценки ресурсоэффективности										
1	Удобство эксплуатации	0,19	4	5	4	4	0,76	0,95	0,76	0,76
2	Надёжность	0,11	5	4	5	5	0,95	0,76	0,95	0,95
3	Требуемые ресурсы	0,2	3	3	2	4	0,57	0,57	0,38	0,76
4	Функциональность	0,19	4	1	4	5	0,76	0,19	0,76	0,95
5	Простота эксплуатации	0,17	5	5	3	4	0,95	0,95	0,57	0,76
Экономические критерии оценки эффективности										
6	Обслуживание	0,07	4	5	4	3	0,76	0,95	0,76	0,57
7	Срок разработки	0,07	4	5	4	2	0,76	0,95	0,76	0,38
	Итого	1	29	28	26	27	5,51	5,32	4,94	5,13

Исходя из проведенного анализа можно заключить, что уязвимость конкурентных технологических решений связана, прежде всего с требуемыми ресурсами и функциональностью.

Однако стоит отметить преимущество в сборе статистических данных над другими конкурирующими разработками в сегменте инструментальных систем.

Наиболее сильным конкурентом можно считать программу Hugin, однако она рассчитана на десктопную версию и интегрировать ее на другие устройства или веб-приложение невозможно, также отсутствует возможность корректировки метрик для сшивки изображений в панорамные снимки.

5.1.3 Диаграмма Исикавы

Диаграмма причины-следствия Исикавы. – графический способ исследования и определения наиболее существенных причинно-следственных взаимосвязей между факторами и последствиями в исследуемой ситуации или проблеме. Причинно-следственная диаграмма представлена на рисунке 30.



Рис. 30 – Диаграмма Исикавы

В качестве путей решения данных проблем можно предложить:

- Использование облачных платформ (Microsoft Azure, Google Cloud Platform и пр.) для вычислений и хранения данных.
- Руководителю следует обучить магистранта и делегировать ему часть полномочий и возможно отправить на дополнительное обучение по новым стандартам методов расчета.
- Арендовать оборудование для сбора данных для хроматографического анализа, и вследствие можно будет сделать проверку на точность.

5.1.4 SWOT-анализ

SWOT-анализ применяют для исследования внешней и внутренней среды проекта. Матрица составляется на основе анализа рынка и конкурентных технических решений, и показывает сильные и слабые стороны проекта, возможности и угрозы для разработки. 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. Матрица SWOT представлена в таблице 7.

Таблица 7 – SWOT-анализ

	Сильные стороны	Слабые стороны
	Си1. Широкий функционал	Сл1. Необходимость изменения системы для каждого заказчика
	Си2. Использование для исследований в разработке	Сл2. Перспективные преобразования только цилиндра
	Си3. Система написана с использованием современных решений	Сл3. Необходимость локального размещения
Возможности		
В1. Доработка в связи с пожеланиями	В1Си3. Система уже имеет широкий функционал, поэтому доработать для определенной сферы просто	В1Сл2. Заказчик может попросить сложную модернизацию
В2. Хранение в репозитории на GitHub	В2Си2. Возможность получать обратную связь и пожелания от сторонних разработчиков	В2Сл3. Хранение на стороннем репозитории зависит от его возможностей
В3. Внедрение на сторонние сервисы	В3Си1. Система может быть интегрирована на различные устройства или веб-приложения. В3Си2. Систему можно использовать для исследований на сторонних сервисах	В3Сл1. Необходимость в изучении специфики работы таких сервисов или устройств

Угрозы		
У1. Динамично развивающиеся технологии	У1Си3. Система написана библиотеке OpenCV, которая является первенствующей библиотекой компьютерного зрения для построения панорам	У1Сл1. По мере развития новых версий, могут возникать проблемы совместимости
У2. Проблемы при развертывании	У2Си1. Так как мы хорошо разбираемся в системе, то решения многих проблем уже известны	У2Сл3. Локальные проблемы намного сложнее отлаживать, поэтому можно проводить аудит с представителями компаний-клиентов
У3. Низкая производительность сшивки, высокая нагрузка на аппаратное обеспечение	У3Си1. Систему можно модифицировать, задействуя аппаратное ускорение определенных операций с использованием GPU. У3Си3. Есть значительный потенциал для адаптации и оптимизации	У3Сл3. Скорость работы может зависеть от мощностей целевого аппаратного обеспечения. Для избежания угрозы нужно тестировать на слабых мощностях и различных конфигурациях и устройствах

В рамках SWOT анализа были рассмотрены по три слабых и сильных сторон проекта, а также три возможности и три угрозы. Указаны стратегии, которые будут ликвидировать слабые стороны разработки и бороться с угрозами.

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.

Соотношения параметров представлены в таблице 8.

Таблица 8 – Интерактивная матрица проекта

Сильные стороны проекта				
Возможности		Си1	Си2	Си3
	B1			+
	B2		+	
	B3	+	+	
Слабые стороны проекта				
Возможности		Сл1	Сл2	Сл3
	B1		+	
	B2			+
	B3	+	-	-
Сильные стороны проекта				
Угрозы		Си1	Си2	Си3
	У1			+
	У2	+		
	У3	+		+
Слабые стороны проекта				
Угрозы		Сл1	Сл2	Сл3
	У1			
	У2			+
	У3			+

Самым большим преимуществом данной системы является то, что систему можно интегрировать на сторонние сервисы и доработать в соответствии со сферой заказчика, а недостатком то, что скорость работы может зависеть от мощностей ПО.

5.1.5 Оценка готовности проекта к коммерциализации

На какой бы стадии жизненного цикла не находилась научная разработка полезно оценить степень ее готовности к коммерциализации и выяснить уровень собственных знаний для ее проведения (или завершения).

Для этого заполним форму в таблице 9, содержащую показатели степени проработанности проекта с позиции коммерциализации и компетенциям разработчика научного проекта.

Таблица 9 – Оценка степени готовности проекта к коммерциализации

№ п/п	Наименование	Степень проработанности НП	Уровень знаний у разработчика
1	Определен имеющийся научно-технический задел	4	3
2	Определены перспективные направления коммерциализации научно-технического задела	5	4
3	Определены отрасли и технологии для предложения	5	5
4	Определена форма для представления на рынок	4	3
5	Определены авторы и осуществлена охрана их прав	1	2
6	Оценена стоимость интеллектуальной собственности	1	2
7	Проведены маркетинговые исследования рынков сбыта	1	1
8	Разработан бизнес-план коммерциализации разработки	1	1
9	Определены пути продвижения разработки на рынок	3	4
10	Разработана стратегия (форма) реализации разработки	4	4
11	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	1	1
12	Проработаны вопросы использования услуг инфраструктуры поддержки, получения льгот	1	1
13	Проработаны вопросы финансирования коммерциализации научной разработки	5	5
14	Имеется команда для коммерциализации разработки	4	2
15	Проработан механизм реализации научного проекта	4	5
	ИТОГО БАЛЛОВ	44	43

Таким образом, готовность научной разработки и ее разработчика к коммерциализации является средней, так как значение находится в пределах от 44 до 30, следовательно, в дальнейшем нужно разрабатывать бизнес-план коммерциализации научной разработки, определять пути продвижения на рынок, прорабатывать вопросы финансирования коммерциализации научной разработки и привлекать компетентных специалистов в команду.

5.1.6 Цели и результаты проекта

Перед определением целей необходимо перечислить заинтересованные стороны проекта. Информация по заинтересованным сторонам представлена в таблице 10:

Таблица 10 – Заинтересованные стороны проекта

Заинтересованные стороны	Ожидания
Пользователь	Простота в использовании программного продукта
Компания пользователя	Облегчен процесс контроля за предприятием
Разработчик	Получение заработной платы
Компания разработчика	Получение прибыли с полноценного комплекса продукта
Научный руководитель, инженер	Выполненная выпускная квалификационная работа

Цели и результат проекта представлены в таблице 11.

Таблица 11 – Цели и результат проекта

Цели проекта	<ul style="list-style-type: none">• Изучить систему• Выбрать средства разработки• Разработать архитектуру системы• Установить и настроить необходимые библиотеки• Создать алгоритмы для сшивки в панорамы• Реализовать код для бесшовной сшивки• Реализовать код для перспективного преобразования
Ожидаемые результаты	<ul style="list-style-type: none">• Успешное развертывание в репозитории полученной панорамы• Сдана выпускная квалификационная работа
Критерии приёмки	<ul style="list-style-type: none">• Успешное тестирование функционала в соответствии с функциональным требованием
Требования к результату проекта	<ul style="list-style-type: none">• Выполненные все пункты функционального требования.• Разработанный функционал полностью соответствует проектным решениям.

Таблица 13 – Расчет затрат на «Спецоборудование для научных работ»

№	Наименование оборудования	Кол-во	Цена, руб.	Стоимость, руб.
1.	Персональный компьютер	1	-	-
2.	Среда разработки PyCharm	1	-	-
3.	Штатив Rekam LightPod RT-L32G	1	1990	1190
4.	Смартфон	1	-	-
Итого				1190

5.2.2.2 Основная заработная плата

В настоящую статью включается основная заработная плата научных и инженерно-технических работников, рабочих макетных мастерских и опытных производств, непосредственно участвующих в выполнении работ по данной теме.

Должность руководителя – доцент, к.т.н. – 35 111,5 рублей в месяц

Должность инженера – студент – 19254 рублей в месяц

Расчет основной заработной платы сводится в таблице 14.

Таблица 14 – Расчет основной заработной платы

№ п/п	Наименование этапов	Исп	Трудоемкость, чел.-дн.	Зарплата, чел.-дн., руб	Всего руб.
1		Р	5	35111,5	35111,5
2		И	84	19254	19254

$$C_{зп} = Z_{осн} + Z_{доп}, \quad (1)$$

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

$Z_{доп}$ – дополнительная заработная плата.

Основная заработная плата $Z_{осн}$ руководителя рассчитывается по следующей формуле:

$$Z_m = Z_{дн} \cdot T_{раб} \quad (2)$$

где $T_{раб}$ – продолжительность работ, выполняемых научно-техническим работником, рабочие дни. (таблица 11);

$Z_{дн}$ – среднедневная заработная плата работника, руб.

Значит, для руководителя:

$$Z_m = 35111 \cdot 1,3 = 45644 \text{ рубля}$$

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{дн} = (Z_m \cdot M) / F_d \quad (3)$$

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 45 раб. дней $M=10,4$ месяца, 6-дневная неделя;

F_d – действительный годовой фонд рабочего времени научно-технического персонала (в рабочих днях) (таблица 14). Тогда,

Для руководителя:

$$Z_{\text{дн}} = \frac{45644 * 10,4}{254} = 1\ 869 \text{ рублей}$$

Для инженера:

$$Z_{\text{дн}} = \frac{19254 * 11,2}{217} = 994 \text{ рубля}$$

Баланс рабочего времени представлен в таблице 15.

Таблица 15 – Баланс рабочего времени

Показатели рабочего времени	Руководитель	Инженер
Календарное число дней	365	365
Количество нерабочих дней		
- выходные дни	52	106
- праздничные дни	14	14
Потери рабочего времени		
- отпуск	56	24
- невыходы по болезни	–	–
Действительный годовой фонд рабочего времени	243	221

Таблица 16 – Результаты расчета основной заработной платы

Исполнители	Z_b , руб.	k_p	Z_m , руб	$Z_{\text{дн}}$, руб.	T_p , дн.	$Z_{\text{осн}}$, руб.
Руководитель	35 111,5	1.3	45644	1869	5	9345
Инженер	19254	1.3	25030	994	84	83496
Итого по статье $Z_{\text{осн}}$:			92841			

5.2.2.3 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:

$$Z_{\text{доп}} = k_{\text{доп}} * Z_{\text{осн}} \quad (4)$$

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

$k_{\text{доп}}$ – коэффициент дополнительной зарплаты ($k_{\text{доп}} = 0,12$);

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

В таблице 17 приведен расчёт основной и дополнительной заработной платы.

Таблица 17 – Расчёт дополнительной заработной платы

Исполнитель	$k_{\text{доп}}$	$Z_{\text{осн}}$	$Z_{\text{доп}}$
Научный руководитель	0,12	9345	1121
Инженер		83496	10019
Итого			11140

5.2.2.4 Отчисления на социальные нужды

Статья включает в себя отчисления во внебюджетные фонды.

$$C_{\text{внеб}} = k_{\text{внеб}} \cdot (Z_{\text{осн}} + Z_{\text{доп}}) \quad (5)$$

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

Величина отчислений во внебюджетные фонды определяется в размере 30% от суммы основной и дополнительной заработной платы.

Для руководителя - 3140 руб.

Для инженера - 28055 руб.

Итого 31195 рублей

5.2.2.5 Накладные расходы

В эту статью 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 16% от суммы основной и дополнительной заработной платы, работников, непосредственно участвующих в выполнении темы.

Расчет накладных расходов ведется по следующей формуле:

$$C_{\text{накл}} = K_{\text{накл}} * (Z_{\text{осн}} + Z_{\text{доп}}) \quad (6)$$

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

$$C_{\text{накл}} = 0,16 * (103981) = 16637 \text{ руб.}$$

5.2.2.6 Формирование бюджета затрат научно-исследовательского проекта

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.

Таблица 18 – Бюджет затрат НТИ

Сырье	Оборудован	Осн зп	Доп зп	Накл рас	Отчисл на соц	Итого
-	1190	92841	11140	16637	31195	153003

В результате было получено, что бюджет на разработку НТИ составит 153003 руб.

5.3 Оценка сравнительной эффективности исследования

5.3.1 Интегральный показатель финансовой эффективности

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:

$$I_{\text{финр}}^{\text{исп}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}}, \quad (7)$$

где $I_{\text{финр}}^{\text{исп}i}$ – интегральный финансовый показатель разработки;

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

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

$$I_{\Phi}^p = \frac{\Phi_i^p}{\Phi_{\text{max}}} = \frac{153003}{170000} = 0,9$$

$$I_{\Phi}^a = \frac{\Phi_i^a}{\Phi_{\text{max}}} = \frac{151813}{170000} = 0,89$$

$$I_{\Phi}^b = \frac{\Phi_i^b}{\Phi_{\text{max}}} = \frac{135064}{170000} = 0,79$$

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в разгах (значение больше единицы), либо соответствующее численное удешевление стоимости разработки в разгах (значение меньше единицы, но больше нуля).

5.3.2 Интегральный показатель ресурсоэффективности

Можно определить следующим образом:

$$I_{pi} = \sum a_i \cdot b_i, \quad (8)$$

где I_{pi} – интегральный показатель ресурсоэффективности для i -го варианта исполнения разработки;

a_i – весовой коэффициент i -го варианта исполнения разработки;

b_i^a, b_i^p – бальная оценка i -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Расчет приведен в форме таблице 19.

Таблица 19 – Сравнение характеристик вариантов исполнения проекта

Критерии ПО	Вес	Текущий	Аналог 1	Аналог 2
Удобство эксплуатации	0,19	4	5	4
Надёжность	0,11	5	4	5
Требуемые ресурсы	0,2	3	3	2
Функциональность	0,19	4	1	4
Простота эксплуатации	0,17	5	5	3
Обслуживание	0,07	4	5	4
Срок разработки	0,07	4	5	4
ИТОГО	1	29	28	26
I_T^x		4.08	3.73	3.54
I_ϕ^x		0.9	0.89	0.79
$I_{финр}^x$		4.53	4.19	4.48

$$\mathcal{E}_{cp} = \frac{I_\phi^p}{I_\phi^a} = \frac{4,53}{4,19} = 1,08$$

Таблица 20 – Сравнительная эффективность разработки

№	Показатели	Разработка	Аналог 1
1	Интегральный финансовый показатель разработки	0,9	0,89
2	Интегральный показатель ресурсоэффективности разработки	4.08	3.73
3	Интегральный показатель эффективности	4.53	4.19
4	Сравнительная эффективность вариантов исполнения	1,08	

Сравнение значений интегральных показателей эффективности позволяет судить о приемлемости существующего варианта решения поставленной в магистерской диссертации технической задачи с позиции финансовой и ресурсной эффективности.

Таким образом, определена сравнительная эффективность проекта равной 1.08. Сравнение значений интегральных показателей эффективности позволяет судить о приемлемости существующего варианта решения поставленной в магистерской диссертации технической задачи с позиции финансовой и ресурсной эффективности.

Вывод

В ходе данной работе были рассмотрены потенциальные потребители результатов исследования, так же для анализа конкурентных технических решений. С позиции ресурсоэффективности и ресурсосбережения была составлена оценочная карта сравнения конкурентных технических решений, по результату которой разрабатываемая система имеет лучшие качества.

Далее был сформирован SWOT-анализ, в котором балы описаны сильные и слабые стороны проекта, в выявлении возможностей и угроз для реализации проекта, для выявления соответствия и несоответствия была составлена интерактивная матрица проекта.

Проведена оценка готовности проекта к коммерциализации, которая показала, что перспективность разработки является средней.

В рамках процессов инициации определены внутренние и внешние заинтересованные стороны проекта с их ожиданиями от проекта, цели и результат проекта.

План проекта представлен на диаграмме Ганта, из которого видно какой исполнитель (студент или руководитель) какой вид работ осуществлял и в течении какого количества дней.

В бюджет инженерно-технического проекта занесена стоимость оборудования, которая составила 1190 рублей. Была рассчитана основная и дополнительная заработная плата исполнителей проекта, сумма которых составила 103981 рублей. Вместе с отчислениями на социальные нужды и накладными расходами бюджет проекта получился равным 153003 рублей.

Был рассчитан интегральный финансовый показатель в ходе оценки бюджета затрат трех вариантов исполнения. Наибольший интегральный

показатель реализации задачи равен 0.9. Значение меньше единицы, а значит полученная величина показывает численное удешевление стоимости. Рассчитан интегральный показатель ресурсоэффективности для трех вариантов исполнения 4.08, 3.73 и 3.54. Расчет интегрального показателя эффективности для разработки и аналога позволил сравнить их и рассчитать сравнительную эффективность разработки, которая говорит нам о приемлемости существующего варианта решения поставленной в магистерской диссертации технической задачи с позиции финансовой и ресурсной эффективности.

6 Социальная ответственность

Выпускная квалификационная работа представляет собой сравнительное исследование метрик для создания алгоритма поэтапной сшивки изображений для задачи генерации панорамных изображений. Алгоритм создания панорамного изображения позволяет создать трехмерные модели окружающего пространства, что может быть применено в сферах туризма, картографии, космической промышленности и т.д. Сферы применения разработки очень широки: от нужд обычных пользователей до специализированных задач.

Описание рабочей зоны: разработка программы велась на постоянном рабочем месте непосредственно за компьютером. Основными элементами рабочего места являлись рабочий стол, рабочий стул, монитор, клавиатура и мышь.

6.1 Правовые и организационные вопросы обеспечения безопасности

6.1.1 Специальные правовые нормы трудового законодательства

С 1 января 2021 года более не действуют 111 постановлений Главного государственного санитарного врача РФ. Охрана труда и техника безопасности за компьютером переходят в зону ответственности работодателя и разрабатываются, исходя из специфики работы, размера коллектива, характера области применения и иных факторов.

Согласно СП 2.2.3670-20 «Санитарно-эпидемиологические требования к условиям труда» юридические лица и индивидуальные предприниматели обязаны осуществлять производственный контроль за условиями труда, и разрабатывать и проводить санитарно-противоэпидемические (профилактические) мероприятия, предусмотренные санитарными правилами федерального закона от 30.03.1999 N 52-ФЗ «О санитарно-эпидемиологическом благополучии населения». Также факторы производственной среды и трудового процесса, воздействующие на работника, должны соответствовать гигиеническим нормативам данного закона [32].

Рабочее место должно соответствовать требованиям ГОСТ 12.2.032-78 «Система стандартов безопасности труда. Рабочее место при выполнении работ сидя». Рабочий стол должен быть устойчивым, иметь однотонное неметаллическое покрытие, не обладающее способностью накапливать статическое электричество; рабочий стул должен иметь дизайн, исключающий онемение тела из-за нарушения кровообращения при продолжительной работе на рабочем месте; рабочее место должно соответствовать техническим требованиям и санитарным нормам [33].

6.1.2 Организационные мероприятия при компоновке рабочей зоны

В данном подразделе рассмотрим основные требования по эргономике рабочего места при выполнении работ сидя. Данные требования регламентируются ГОСТ 12.2.032-78 [34]. В ГОСТ указаны такие параметры, как высота сидения, ширина пространства для ног, высота рабочей поверхности для человека, выполняющего работы в сидячем положении.

Исходя из общих принципов организации рабочего места, основными элементами рабочего места программиста являются: рабочий стол, рабочий стул (кресло), монитор, клавиатура, мышь; вспомогательными – пюпитр, подставка для ног.

Согласно ГОСТ 12.2.032-78, взаимное расположение элементов рабочего места должно обеспечивать возможность осуществления всех необходимых движений для эксплуатации и технического обслуживания оборудования. Рабочие места с ПЭВМ должны располагаться на расстоянии не менее 1,5 м от стены с оконными проемами, от других стен – на расстоянии 1 м, между собой – на расстоянии не менее 1,5 м. При размещении рабочих мест необходимо исключить возможность прямой засветки экрана источником естественного освещения. При размещении ЭВМ на рабочем месте должно обеспечиваться пространство для пользователя величиной не менее 850 мм.

Конструкция рабочего стола должна обеспечивать возможность размещения на рабочей поверхности необходимого комплекта оборудования и документов с учетом характера выполняемой работы.

Рабочие столы по конструктивному исполнению подразделяют на регулируемые и нерегулируемые по изменению высоты рабочей поверхности.

В ГОСТ 22269-76 «Рабочее место оператора. Взаимное расположение элементов рабочего места» описаны общие требования о расположении элементов рабочего места. Взаимное расположение элементов рабочего места должно обеспечивать возможность осуществления всех необходимых движений и перемещений для эксплуатации и технического обслуживания оборудования. Взаимное расположение элементов рабочего места должно обеспечивать необходимые зрительные и звуковые связи между оператором и оборудованием, а также между операторами.

Независимо от профессионального назначения есть общие требования для сидений длительного пользования:

1. Сиденье должно обеспечивать позу, способствующую уменьшению статической работы мышц.
2. Сиденье в целом и его элементы должны создавать условия для возможности изменения рабочей позы.
3. Наличие ряда регулируемых параметров (высота сиденья, угол наклона спинки, высота спинки) [34].

Поверхность сиденья должна иметь ширину и глубину не менее 400 мм. Должна быть предусмотрена возможность изменения угла наклона поверхности сиденья от 15° вперед до 5° назад. Высота поверхности сиденья должна регулироваться в пределах от 400 до 550 мм. Опорная поверхность спинки стула (кресла) должна иметь высоту (300 ± 20) мм, ширину не менее 380 мм и радиус кривизны в горизонтальной плоскости 400 мм. Угол наклона спинки в вертикальной плоскости должен регулироваться в пределах $0^\circ\pm 30^\circ$ от вертикального положения. Расстояние спинки от переднего края сиденья должно регулироваться в пределах от 260 до 400 мм.

Подлокотники должны быть длиной не менее 250 мм, шириной – 50-70 мм, иметь возможность регулирования по высоте над сиденьем в пределах

(230±30) мм и регулирования внутреннего расстояния между подлокотниками в пределах от 350 до 500 мм.

Согласно ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения», дисплей на рабочем месте оператора должен располагаться так, чтобы изображение в любой его части было различимо без необходимости поднять или опустить голову (рис. 31) [35].

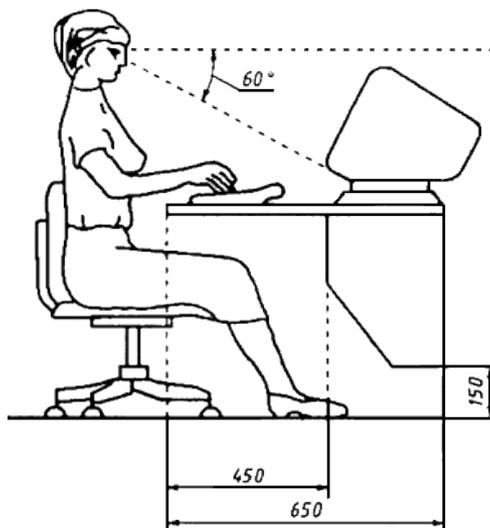


Рис. 31 – Расположение дисплея на столе

Дисплей на рабочем месте должен быть установлен ниже уровня глаз оператора. Угол наблюдения экрана оператором относительно горизонтальной линии взгляда не должен превышать 60° , как показано на рисунке 1.

Расстояние от глаз пользователя до экрана монитора должно быть не менее 50 сантиметров, оптимально – 60-70 сантиметров. Расстояние от экрана монитора до задней стенки монитора соседнего ряда должно быть не менее 2 метров, а расстояние между боковыми стенками – не менее 1,2 метров.

Клавиатуру следует располагать на поверхности стола на расстоянии 100-300 мм от переднего края, обращенного к пользователю или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

Пюпитр должен быть установлен на одном уровне с экраном дисплея и отстоять от глаз оператора на том же расстоянии, что и экран, либо отличаться

от него, но не более чем на 100 мм. Поверхность пюпитра должна иметь покрытие из диффузно отражающего материала с коэффициентом отражения 0,45-0,50.

Подставка для ног должна регулироваться по высоте в пределах до 150 мм и углу наклона опорной поверхности – до 20°. Ширина опорной поверхности подставки для ног должна быть не менее 300 мм, глубина - не менее 400 мм. Поверхность подставки должна быть рифленой. По переднему краю должен быть предусмотрен бортик высотой 10 мм.

Помещения с компьютерами в обязательном порядке должны быть оборудованы системами эффективной приточно-вытяжной вентиляцией отопления и кондиционирования воздуха. Внутренняя отделка интерьера помещений с компьютерами должна быть сделана при использовании диффузно-отражающих материалов. В обязательном порядке в помещении должны находиться углекислотный огнетушитель для тушения пожара и аптечка первой медицинской помощи.

6.2 Производственная безопасность

Вредные и опасные факторы, воздействующие на сотрудника, устанавливаются согласно ГОСТ 12.0.003-2015 «Опасные и вредные производственные факторы. Классификация» [36].

В ходе выполнения работы и при дальнейшей эксплуатации разработанного программного средства могут возникнуть следующие вредные и опасные факторы (табл. 21).

Таблица 21 – Возможные опасные и вредные факторы

Факторы (ГОСТ 12.0.003-2015)	Этапы работ			Нормативные документы
	Разработка	Эксплуатация	Обслуживание	
Отклонение показателей микроклимата	+	+	+	– СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания. – СанПиН 2.2.4.548-96. Гигиенические требования к микроклимату производственных помещений.
Превышение уровня шума		+		– СН 2.2.4/2.1.8.562-96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории застройки. Санитарные нормы. – СП 51.13330.2011. Свод правил. Защита от шума.
Недостаточная освещенность рабочей зоны	+	+	+	– СП 52.13330.2016 Свод правил. Естественное и искусственное освещение. – ГОСТ Р 55710-2013 Освещение рабочих мест внутри зданий. Нормы и методы измерений
Психофизические факторы (монотонность труда, нервно-психические перегрузки, перенапряжение зрительных анализаторов)	+			Приказ Минтруда России от 24.01.2014 N 33н (ред. от 27.04.2020) Об утверждении Методики проведения специальной оценки условий труда, Классификатора вредных и (или) опасных производственных факторов, формы отчета о проведении специальной оценки условий труда и инструкции по ее заполнению.
Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека	+	+	+	ГОСТ 12.1.028-82 ССБТ. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов.

6.2.1 Анализ вредных и опасных факторов

6.2.1.1 Отклонение показателей микроклимата рабочей зоны

Микроклимат производственных зданий описывает следующие параметры: скорость движения воздуха, относительная влажность, температура воздуха в помещении и поверхностей. Эти факторы влияют на организм человека, определяя его самочувствие.

Допустимые и оптимальные значения параметров микроклимата приведены в таблице 22. В данном случае работа относится к категории труда «1б». Данные были скорректированы в соответствии с оптимальными микроклиматическими условиями для рабочего места оператора по ГОСТ Р 50923-96.

Таблица 22 – Оптимальные и допустимые нормы температуры, относительной влажности и скорости движения воздуха в рабочей зоне оператора [38].

Период года	Категория работ по уровню энергозатрат	Температура, °С			Температура поверхностей, °С		Относительная влажность, %		Скорость движения воздуха, м/с	
		Оптимальная	Допустимая на рабочих местах		Оптимальная	Допустимая	Оптимальная	Допустимая	Оптимальная, не более	Допустимая, не более
			ниже оптимальных величин	выше оптимальных величин						
Холодный	1б (140-174 Вт)	21 – 23	19,0 – 20,9	23,1 – 24,0	20,0 – 24,0	18,0 – 25,0	40 – 60	15 – 75	0,1	0,2
Теплый	174 Вт)	23 – 25	20,0 – 21,9	24,1 – 28,0	21,0 – 25,0	19,0 – 29,0	40 – 60	15 – 75	0,1	0,3

Меры по улучшению воздушной среды в производственном помещении включают: правильную организацию вентиляции и кондиционирования воздуха, отопление помещений. Вентиляция может осуществляться естественным и механическим способом.

В здании должны подаваться следующие объёмы наружного воздуха: при объёме помещения более 45 м³ на человека – не менее 35 м³ в час на человека;

при объёме помещения до 25 м³ на человека и отсутствии выделения вредных веществ разрешается естественная вентиляция.

В зимнее время в помещении должна быть предусмотрена система водяного отопления со встроенными нагревательными элементами и терморегуляторами.

6.2.1.2 Повышенный уровень шума на рабочем месте

Шум создает значительную нагрузку на нервную систему человека, оказывая на него психологическое воздействие.

Шумовой фон помещения создают десять одновременно работающих компьютеров. Также возникает шум, исходящий от принтера или телефонных аппаратов. Также источником шума является система вентиляции или шумы, поступающие извне помещения.

Уровень звука на рабочих местах, связанных с творческой деятельностью, научной деятельностью, программированием, преподаванием и обучением не должен превышать 50 дБА согласно СН 2.2.4/2.1.8.562-96 [39].

Предельно допустимые и допустимые уровни звукового давления, для проникающего шума следует принимать по таблице 23 согласно СП 51.13330.2011 [40].

Таблица 23 – Предельно допустимые и допустимые уровни звукового давления по СП 51.13330.2011

Уровни звукового давления (эквивалентные уровни звукового давления), дБ, в октавных полосах частот со среднегеометрическими частотами, Гц										Уровень звука L _A (эквивалентный уровень звука L _{Aэкв}), дБА	Максимальный уровень звука L _{Amax} , дБА
Гц	31,5	63	125	250	500	1000	2000	4000	8000		
дБ	86	71	61	54	49	45	42	40	38	50	65

Постоянный проникающий шум считают удовлетворяющим нормам, если уровни звукового не превышают предельно допустимых и допустимых уровней звукового давления и допустимых уровней звука, указанных в таблице 22.

Меры, которые необходимо принять, для того чтобы помещение было менее зашумленным – это обеспечить нормальную вентиляцию системного блока. Для охлаждения необходимо оборудовать со стороны вентиляционных отверстий хотя бы 20-30 см свободного пространства.

6.2.1.3 Недостаточная освещенность рабочей зоны

При длительной работе плохая освещенность рабочей зоны приводит к ухудшению зрения и к ослаблению зрительной активности работающего.

Рабочее место разработчика согласно санитарно-гигиеническим требованиям может освещаться искусственным, естественным или комбинированным освещением [41]. В СП 52.13330.2016 описаны требования к освещению помещений промышленных предприятий (табл. 24).

Таблица 24 – Требования к освещению помещений промышленных предприятий при искусственном освещении

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта	Разряд зрительной работы	Подразряд зрительной работы	Контраст объекта с фоном	Характеристика фона	Искусственное освещение		
						Освещенность, ЛК		
						При комб. освещении		При общ. освещении
						всего	в т.ч. от общего	
Высокой точности	от 0,3 до 0,5	III	Г	Средний Большой “	Светлый “ Средний	400	200	200

Также в ГОСТ Р 55710-2013 «Освещение рабочих мест внутри зданий. Нормы и методы измерений» минимально допустимое значение освещенности для компьютерных классов в учебных учреждениях – 300 ЛК [42].

Для выявления соответствия аудитории №103 10 корпуса ТПУ требованиям СП 52.13330.2016 и ГОСТ Р 55710-2013 далее будут приведены расчеты искусственного освещения рабочей зоны работника согласно методическим указаниям к выполнению лабораторной работы «Расчет искусственного освещения» для студентов дневного и заочного обучения всех направлений и специальностей ТПУ [43].

Параметры аудитории №103 10 корпус ТПУ:

- площадь $S = 44,9$ м².;
- длина $A = 9,2$ м.;
- ширина $B = 4,88$ м.;
- высота потолка $H = 2,5$ м.;
- высота рабочей поверхности $h_{rp} = 0,8$ м.;
- количество светильников $n = 14$;
- тип светильников ЛВО 4×18 CSVТ люминесцентными лампами типа L18W/640 с потоком $F = 1200$ лм.

Согласно отраслевым нормам освещенности, уровень рабочей поверхности над полом составляет 0,8 м., а установлена минимальная норма освещенности $E = 300$ ЛК [44].

Расчетная высота подвеса светильников над рабочей поверхностью (h) определяется по формуле (1):

$$h = H - h_p - h_c \quad (1)$$

где H – высота потолка в помещении, м;

h_p – расстояние от пола до рабочей поверхности стола, м; h_c – расстояние от потолка до светильника, м;

Так как светильники установлены в уровень с потолком, примем h_c равным 0. Таким образом, согласно формуле (1), высота подвеса светильников над рабочей поверхностью для аудитории №103 корпуса №10 ТПУ равна:

$$h = 2,5 - 0,8 = 1,7 \text{ м}$$

Индекс помещения определяется по формуле (2):

$$i = \frac{S}{h \cdot (A + B)} \quad (2)$$

где S – площадь помещения, м²; A – длина комнаты, м;

B – ширина комнаты, м;

h – высота подвеса светильников, м.

Индекс помещения для аудитории №103 корпуса №10 ТПУ:

$$i = \frac{44,9}{1,7 \cdot (9,2 + 4,88)} = 1,776$$

Исходя из того, что потолок в аудитории представлен светлыми потолочными панелями, поверхность стен окрашена белой краской и имеется два окна без штор, примем коэффициенты отражения от стен $\rho_c = 50\%$ и потолка $\rho_{п} = 50\%$.

По таблице коэффициентов использования светового потока из пособия к МГСН 2.06-99 для соответствующих значений i , ρ_c , $\rho_{п}$ примем $\eta = 0,36$ (для потолочного светильника любого типа при $i = 1,75$) [44].

Освещенность помещения рассчитывается по формуле (3):

$$E_{\phi} = \frac{n \times \eta \times \Phi}{S \times k_3} \quad (3)$$

где Φ – световой поток светильника, лм;

S – площадь помещения, m^2 ;

k_3 – коэффициент неравномерности освещения;

n – число светильников;

η – коэффициент использования светового потока.

Коэффициент запаса k учитывает запыленность светильников и их износ. Для помещений с малым выделением пыли $k = 1,5$. Поправочный коэффициент z это коэффициент неравномерности освещения. Для люминесцентных ламп $z = 1,1$. Каждый светильник состоит из 4 ламп.

Учитывая все параметры, освещенность равна:

$$E_{\phi} = \frac{14 \times 4 \times 0,36 \times 1200}{44,9 \times 1,5 \times 1,1} = 326,5 \text{ лк} \quad (4)$$

Согласно ГОСТ Р 55710-2013 освещенность компьютерной аудитории №103 корпуса №10 ТПУ соответствует норме, если руководствоваться СП 52.13330.2016 – превышает норму.

6.2.1.4 Нервно-психические перегрузки, связанные с напряженностью трудового процесса

Работа с ПК сопряжена с воздействием вредных психофизиологических факторов, в частности, нервно-психических перегрузок. Нервно-психические

перегрузки организма работающего, связанные с напряженностью трудового процесса характеризуются такими показателями, как:

- длительность сосредоточенного наблюдения;
- плотность сигналов (световых, звуковых) и сообщений в единицу времени;
- число производственных объектов одновременного наблюдения;
- нагрузка на слуховой анализатор;
- активное наблюдение за ходом производственного процесса;
- работа с оптическими приборами;
- нагрузка на голосовой аппарат [36].

Отнесение условий труда по напряженности трудового процесса осуществляется при наблюдении в среднем за час работы при необходимости на основе хронометражных наблюдений.

В Приложении N 21 к Приказу Минтруда России от 24 января 2014 г. N 33н отнесены условия труда по классу (подклассу) условий труда, связанные с напряженностью трудового процесса. Класс (подкласс) условий труда устанавливается по показателю напряженности трудового процесса, имеющему наиболее высокий класс (подкласс) условий труда [45].

При работе с ПЭВМ основным фактором, влияющим на нервную систему программиста или пользователя, является огромное количество информации, которое он должен воспринимать. Это является сложной задачей, которая очень сильно влияет на сознание и психофизическое состояние из-за монотонности работы.

Организация работы с ПЭВМ осуществляется в зависимости от вида и категории трудовой деятельности. Виды трудовой деятельности разделяются на 3 группы: группа А – работа по считыванию информации с экрана с предварительным запросом; группа Б – работа по вводу информации; группа В – творческая работа в режиме диалога с ПЭВМ. Работа программиста-разработчика рассматриваемой в данной работе системы относится к группам Б и В. Категории трудовой деятельности различаются по степени тяжести выполняемых работ. Для снижения воздействия рассматриваемого вредного

фактора предусмотрены регламентированные перерывы для каждой группы работ – таблица 25 [46].

Таблица 25 – Суммарное время регламентированных перерывов в зависимости от продолжительности работы, вида категории трудовой деятельности с ПЭВМ

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

Согласно таблице, требуется установить перерывы, сумма которых за смену составит не менее 70 или 90 минут в соответствии с группой при 8-часовой смене.

При работе за ПК оператор читает текст в прямом свете, его глаза смотрят на источник света. При работе за дисплеем возникает следующая проблема: экран имеет высокую фоновую яркость, при ее понижении уменьшается контрастность изображения, что создает проблему для считывания информации. Поэтому пользователю приходится повышать яркость, что приводит к увеличению интенсивности вредных излучений и к утомлению глаз.

В настоящее время при работе за ПК визуальные параметры мониторов оказывают большую нагрузку на зрительные анализаторы, а именно на мышцы глаз. После длительной нагрузки необходим период восстановления и отдыха.

6.2.1.5 Повышенное значение напряжения в электрической цепи

Несоблюдение правил ГОСТ 12.1.038-82 может привести к опасным последствиям [47]. Поражение электрическим током может произойти при прикосновении к токоведущим частям, находящимся под напряжением, на которых остался заряд или появилось напряжение.

Электрический ток оказывает на человека термическое, электролитическое, биологическое и механическое воздействие. Действие электрического тока на человека приводит к травмам или гибели людей. Для переменного тока частотой 50 Гц допустимое значение (не более) напряжения прикосновения составляет 2 В, а силы тока – 0,3 мА, для тока частотой 400 Гц, соответственно – 3 В и 0,4 мА, для постоянного тока – 8 В и 1 мА. Мерами защиты от воздействия электрического тока являются оградительные устройства, устройства автоматического контроля и сигнализации, изолирующие устройства и покрытия, устройства защитного заземления, устройства автоматического отключения, предохранительные устройства.

Токи статического электричества, наведенные в процессе работы компьютера на корпусах монитора, системного блока и клавиатуры, могут приводить к разрядам при прикосновении к этим элементам. Такие разряды опасности для человека не представляют, но могут привести к выходу из строя компьютера. Для снижения величин токов статического электричества используются нейтрализаторы, местное и общее увлажнение воздуха, использование покрытия полов с антистатической пропиткой.

6.2.2 Обоснование мероприятий по защите исследователя от действия опасных и вредных факторов

Разрабатываемый в ходе выполнения ВКР программная система используется для анализа данных для автоматической сшивки изображений и их сопоставления. Большую часть при работе с программной системой оператору необходимо контролировать параметры входных данных и собирать выходные панорамные снимки. Следовательно, преимуществом данного продукта является практически полная автономность.

В качестве помещения для исследования необходимо выбрать сухое, хорошо отапливаемое помещение с непроводящими ток полами, с температурой 18-21° и влажностью 40-50 %.

В качестве дополнительных мер безопасности, необходимо проверять отсутствие проводов на рабочей поверхности и отсутствие влаги.

6.3 Экологическая безопасность

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

Непосредственно программный продукт, разработанный в ходе выполнения магистерской диссертации, не наносит вреда окружающей среде ни на стадиях его разработки, ни на стадиях эксплуатации.

Обращение с отходами регламентируется ГОСТ Р 53692-2009 [48]. Поэтому, при необходимости утилизировать вышедшую из употребления электронику наиболее безопасным для окружающей среды способом необходимо обращаться в специализированную компанию по утилизации. Такие компании действуют на всей территории РФ, в том числе и в Томской области. Необходимо отметить, что в целом при работе с компьютером существенного загрязнения окружающей среды не происходит и вредные выбросы не сравнимы с производственными.

Современные ПЭВМ производят практически без использования вредных веществ, опасных для человека и окружающей среды. Исключением являются аккумуляторные батареи компьютеров и мобильных устройств (код отхода 920 110 01 53 2 или 920 110 01 52 3, класс опасности – 2 или 3 [49,50]). В аккумуляторах содержатся тяжелые металлы, кислоты и щелочи, которые могут наносить ущерб окружающей среде, попадая в гидросферу и литосферу, если они были неправильно утилизированы. Для утилизации аккумуляторов необходимо обращаться в специальные организации, специализировано занимающиеся приемом, утилизацией и переработкой аккумуляторных батарей.

Люминесцентные лампы (код отхода 471 101 01 52 1, класс опасности – 1 [49, 50]), 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.

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.

Бытовой мусор помещений организаций несортированный, образованный в результате деятельности работников предприятия (код отхода 733 100 02 72 5, класс опасности – 5 [49, 50]). Агрегатное состояние отхода твердое; основные компоненты: бумага и древесина, металлы, пластмассы и др. Для сбора мусора рабочее место оснащается урной. При заполнении урны, мусор выносится в контейнер бытовых отходов. Предприятие заключает договор с коммунальным хозяйством по вывозу и размещению мусора на организованных свалках.

Разработанный алгоритмом может участвовать в информировании населения о местах свалок и многих природных комплексах, связанных с массовой гибелью живых организмов.

6.4 Безопасность в чрезвычайных ситуациях

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 (СП 12.13130.2009) относится к категории Д (из 5-ти категорий А, Б, В1-В4, Г, Д), т.к. относится к помещениям с негорючими веществами и материалами в холодном состоянии [51].

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

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».

Рабочее помещение 103 аудитории в 10 корпусе оборудовано в соответствии с требованиями пожарной безопасности. Имеется порошковый огнетушитель, а также пожарная сигнализация и средства связи. В случае возникновения пожара в здании автоматически срабатывают датчики пожаротушения, и звуковая система оповещает всех сотрудников о немедленной эвакуации из здания, которые должны направиться на выход в соответствии с планом эвакуации (рис. 32).



Рис. 32 – План эвакуации при пожаре и других ЧС

Прототип разработанного алгоритма был представлен на Онлайн-интенсиве по искусственному интеллекту и анализу данных «Архипелаг 20.35» для локализации лесных пожаров, однако может быть использован и в других областях.

Разработанный алгоритм может помочь для обеспечения безопасности в чрезвычайных случаях как часть программного продукта. Программы такого рода служат для получения панорамного изображения и применение для проведения следственных действий, оперативных мероприятий и криминалистических экспертиз с целью расследования преступлений и представления в суд наглядного доказательственного материала.

Выводы и рекомендации

Проанализировав условия труда на рабочем месте, где был разработан раздел «Социальная ответственность». В ходе выполнения работы были рассмотрены основные требования при компоновке рабочей зоны, выявлены

опасные и вредные факторы, воздействию которых может подвергнуться человек, использующий алгоритм поэтапной сшивки изображений для задачи генерации панорамных изображений.

Был проведен анализ нормативной документации. В целом, рабочее место удовлетворяет требованиям безопасности. Выполняемая работа не сопряжена с высоким риском травматизма.

Освещение на рабочем месте соответствует нормам – используется несколько энергосберегающих ламп. Уровни шума находятся в допустимых пределах – источником шума при эксплуатации ПК могут являться системы охлаждения, а также жесткий диск, однако уровень создаваемого ими шума не высок. Микроклиматические условия соблюдаются за счет использования систем отопления и кондиционирования. Защита от повреждений электроники статическим электричеством не обеспечивается, однако так как корпус ПК закрыт, вероятность поражения элементов или работника минимальна (если не прикасаться мокрыми руками к корпусу). Во время работы делаются перерывы для снижения нагрузки и предотвращения нервно-психических перегрузок. Помещение оборудовано согласно требованиям электробезопасности. В случае выхода из строя используемой электроники или ламп, отходы передаются в соответствующие компании.

Работа с ЭВМ можно отнести к экологически безопасным видам деятельности, если должным образом утилизировать отходы данной деятельности. Относительно рассмотренного вопроса об экологической безопасности можно сказать, что деятельность в ходе выполнения выпускной квалификационной работы не представляет опасности окружающей среде.

Рабочее помещение оборудовано в соответствии с требованиями пожарной безопасности. Имеется порошковый огнетушитель, а также пожарная сигнализация.

Заключение

В результате работы над выпускной квалификационной работой были предложены и разработаны несколько алгоритмов: сортировка изображений, смешение изображений, цветовая коррекция, сшивка пары изображений, цилиндрическая проекция, создание полноценной панорамы, а также визуализация готовой панорамы с использованием веб-технологий.

В процессе исследования и реализации были рассмотрены и исследованы существующие методы поиска ключевых точек: SIFT, ORB, BRISK, AKAZE, а также методы их сопоставления (BFMatcher, FLANN) и метрики (расстояния Хэмминга, Евклида, Манхэттена, k-средних).

Для реализации самой сшивки были рассмотрены аффинные преобразования и гомография, функции соответствия RANSAC, LMEDS, PROSAC.

В результате проведённого исследования можно заключить, что метод поиска особенностей BRISK обладает самой высокой производительностью – больше всего найденных ключевых точек на тестовых изображениях и наименьшее среднее время поиска. Лучший способ сопоставления для него - BFMatcher с функцией расстояния Хэмминга и вероятностным методом RANSAC, поскольку эта комбинация даёт максимальное количество точек, по которым будет производиться сшивка. Это всё позволяет предположить, что данный метод может быть достаточно эффективным для сшивки кадров в панорамные изображения, в том числе и в реальном времени. В то же время, результаты могут варьироваться на разных парах изображений и нельзя исключать, что в каких-то ситуациях будут предпочтительны другие методы.

В свою очередь вероятностный метод RANSAC показал себя наиболее эффективным в условиях исследования на тестовой выборке изображений.

Проведённая работа в целом, собранные в её процессе статистические данные и их анализ, могут быть полезны для принятия решений в процессе разработки алгоритмов и реализации приложений, основанных на компьютерном зрении в самых различных областях применения, например,

виртуальные туры, аэрофотосъемка и картографирование. То есть везде, где может быть задействована сшивка изображений и нахождение ключевых особенностей.

Список публикаций студента

1. Унжакова Е.В., Друки А.А. Разработка алгоритма поэтапного совмещения перекрывающихся изображений для сравнительного анализа методов обнаружения ключевых точек // Молодежь и современные информационные технологии: сборник трудов XVIII Международной научно-практической конференции студентов, аспирантов и молодых ученых, Томск, 22–26 марта 2021 г. – Томск: ТПУ, 2021 – Принята к печати.

2. Унжакова Е.В., Друки А.А. Сравнительный анализ методов обнаружения особых точек на перекрывающихся изображениях общей области пространства // Молодежь и современные информационные технологии: сборник трудов XVIII Международной научно-практической конференции студентов, аспирантов и молодых ученых, Томск, 22–26 марта 2021 г. – Томск: ТПУ, 2021 – Принята к печати.

Список используемых источников

1. How to Take 360° Photos at a Fixed NPP and Manually Stitch Them [Электронный ресурс]. – URL: <https://medium.com/cupix/how-to-take-360-photos-at-a-fixed-npp-and-manually-stitch-them-6abcb7af2cb7> (дата обращения: 17.12.2020).
2. Szeliski, R., 2011. Computer Vision-Algorithms and Applications. 3, pp: 181-228.
3. I. R. Otero, M. Delbracio. Anatomy of the SIFT Method [Электронный ресурс]. – URL: <https://www.ipol.im/pub/art/2014/82/article.pdf> (дата обращения: 17.02.2020).
4. H. Bay, A. Ess, T. Tuytelaars, L. Van Gool. Speeded Up Robust Features [Электронный ресурс]. – URL: <https://www.cs.jhu.edu/~misha/ReadingSeminar/Papers/Bay08.pdf> (дата обращения: 10.02.2021).
5. Обнаружение устойчивых признаков изображения: метод SURF [Электронный ресурс]. – URL: <http://habrahabr.ru/post/103107/> (дата обращения: 18.02.2020).
6. E. Rublee, V. Rabaud, K. Konolige, G. Bradski. ORB: an efficient alternative to SIFT or SURF [Электронный ресурс]. – URL: https://www.researchgate.net/publication/221111151_ORB_an_efficient_alternative_to_SIFT_or_SURF (дата обращения: 18.02.2020).
7. ORB (Oriented FAST and Rotated BRIEF) [Электронный ресурс]. – URL: http://docs.opencv.org/trunk/doc/py_tutorials/py_feature2d/py_orb/py_orb.html (дата обращения: 18.02.2020).
8. Stefan Leutenegger, Margarita Chli, Roland Siegwart: «BRISK: Binary Robust Invariant Scalable Keypoints». Computer Vision (ICCV), pp. 2548 –2555, 2011.
9. S. Grewenig, J. Weickert, C. Schroers, A. Bruhn: «Cyclic Schemes for PDE-Based Image Analysis», In International Journal of Computer Vision, 2013.

10. Метод k-средних [Электронный ресурс]. – URL: https://ru.wikipedia.org/Метод_k-средних (дата обращения: 19.02.2020).
11. Алексеева А. А. Математические методы интеллектуального анализа данных и вывода по прецедентам / А. А. Алексеева, М. А. Тараник // Информационные технологии в науке, управлении, социальной сфере и медицине : сборник научных трудов III Международной научной конференции, 23-26 мая 2016 г., Томск : в 2 ч. — Томск : Изд-во ТПУ, 2016. — Ч. 1. — [С. 636-639].
12. Манхэттенское расстояние [Электронный ресурс]. – URL: https://ru.wikipedia.org/wiki/Расстояние_городских_кварталов (дата обращения: 22.02.2020).
13. Раджараман, А., Ульман, Дж. Mining of Massive Datasets [Электронный ресурс]. – URL: <http://infolab.stanford.edu/~ullman/mmds.html> (дата обращения: 15.02.2020).
14. О решении задачи линейного программирования в эпоху больших данных [Электронный ресурс]. – URL: https://www.researchgate.net/publication/320040468_О_решении_задачи_линейного_программирования_в_эпоху_больших_данных/references (дата обращения: 27.03.2020).
15. Richard Szeliski. Computer Vision: Algorithms and Applications / Texts in Computer Science. Springer Science & Business Media, 2011. С. 812.
16. М. А. Fischler and R. C. Bolles, «Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,» Communications of the ACM, vol. 24, no. 6, pp. 381-395, 1981.
17. Least Median of Squares Regression. [Электронный ресурс]. – URL: https://www.researchgate.net/publication/242464484_Least_Median_of_Squares_Regression (дата обращения: 17.04.2020).
18. O. Chum and J. Matas, «Matching with PROSAC-progressive sample consensus,» in Computer Vision and Pattern Recognition, San Diego, CVPR, 2005, pp. 220-226.

19. Проекция панорамных изображений [Электронный ресурс]. – URL: <https://www.cambridgeincolour.com/ru/tutorials-ru/image-projections.htm> (дата обращения: 22.05.2020).
20. CIELAB color space [Электронный ресурс]. – URL: https://en.wikipedia.org/wiki/CIELAB_color_space (дата обращения: 17.01.2021).
21. Stitching and Blending. Kari Pulli [Электронный ресурс]. – URL: <https://web.stanford.edu/class/cs231m/lectures/lecture-5-stitching-blending.pdf> (дата обращения: 02.02.2021).
22. ImageBlending [Электронный ресурс]. – URL: https://piazza.com/class_profile/get_resource/hz5ykuetdmr53k/i0zvj8uvcao7lp (дата обращения: 17.01.2021).
23. VRWorks - 360 Video - Multiband Blending [Электронный ресурс]. – URL: <https://developer.nvidia.com/vrworks/vrworks-360video/multiband-blending> (дата обращения: 17.01.2021).
24. Image Blending Using Laplacian Pyramids [Электронный ресурс]. – URL: <https://becominghuman.ai/image-blending-using-laplacian-pyramids-2f8e9982077f> (дата обращения: 17.01.2021).
25. Документация OpenCV [Электронный ресурс]. – URL: <http://opencv.org/> (дата обращения: 10.02.2021).
26. Feature Matching. OpenCV-Python Tutorials [Электронный ресурс]. – URL: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_matcher/py_matcher.html (дата обращения: 12.02.2021).
27. Edge blending using commodity projectors [Электронный ресурс]. – URL: <http://paulbourke.net/miscellaneous/edgeblend/> (дата обращения: 28.03.2021).
28. High level stitching API (Stitcher class) [Электронный ресурс]. – URL: https://docs.opencv.org/3.4/d8/d19/tutorial_stitcher.html (дата обращения: 10.05.2021).
29. Panorama photo stitcher «Hugin» [Электронный ресурс]. – URL: <https://sourceforge.net/projects/hugin/> (дата обращения: 10.05.2021).

30. A panoramic image stitching tool made with OpenCV and Python [Электронный ресурс]. – URL: <https://github.com/kupiakos/image-stitcher> (дата обращения:10.05.2021).
31. A panoramic image stitching tool made with OpenCV and Python [Электронный ресурс]. – URL: <https://github.com/kupiakos/image-stitcher> (дата обращения:10.05.2021).
32. СП 2.2.3670-20 Санитарно-эпидемиологические требования к условиям труда.
33. ГОСТ 12.2.032-78 Система стандартов безопасности труда. Рабочее место при выполнении работ сидя.
34. ГОСТ 22269-76 Рабочее место оператора. Взаимное расположение элементов рабочего места.
35. ГОСТ Р 50923-96 Дисплей. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения.
36. ГОСТ 12.0.003-2015 Опасные и вредные производственные факторы. Классификация.
37. СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания.
38. СанПиН 2.2.4.548-96. Гигиенические требования к микроклимату производственных помещений.
39. СН 2.2.4/2.1.8.562-96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории застройки. Санитарные нормы.
40. СП 51.13330.2011. Свод правил. Защита от шума.
41. СП 52.13330.2016 Свод правил. Естественное и искусственное освещение.
42. ГОСТ Р 55710-2013 Освещение рабочих мест внутри зданий. Нормы и методы измерений.
43. Безопасность жизнедеятельности. Расчет искусственного освещения. Методические указания к выполнению индивидуальных заданий для студентов

дневного и заочного обучения всех направлений и специальностей ТПУ. – Томск: Издательство Томского политехнического университета, 2008. – 20 с.

44. СП 52.13330.2016 Свод правил. Естественное и искусственное освещение.

45. Приказ Минтруда России от 24.01.2014 N 33н (ред. от 27.04.2020) Об утверждении Методики проведения специальной оценки условий труда, Классификатора вредных и (или) опасных производственных факторов, формы отчета о проведении специальной оценки условий труда и инструкции по ее заполнению.

46. Судебные и нормативные акты РФ. Время регламентированных перерывов в зависимости от продолжительности рабочей смены, вида и категории трудовой деятельности с ВДТ и ПЭВМ [Электронный ресурс]. – URL: <https://sudact.ru/law/pot-ro-14000-005-98-polozhenie-raboty-s-povyshennoi-polozhenie/23/tablitza-20/> (дата обращения: 10.05.2021).

47. ГОСТ 12.1.028-82 ССБТ. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов.

48. ГОСТ Р 53692-2009. Ресурсосбережение. Обращение с отходами.

49. Приказ Росприроднадзора от 22.05.2017 №242. Об утверждении Федерального каталога отходов.

50. ГОСТ 12.1.007-76. Система стандартов безопасности труда (ССБТ). Вредные вещества. Классификация и общие требования безопасности (с Изменениями N 1, 2)

51. СП 12.13130.2009. Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности.

Приложение А Результаты алгоритма для совмещения изображений и получения статистических данных

Таблица 1 Поиск ключевых особенностей на изображениях «MOUNTAIN»

Метод	Количество точек		Время поиска, мс
	Левая картинка	Правая картинка	
SIFT	1490	674	120
ORB	6629	4281	474
BRISK	3485	1313	77
AKAZE	429	260	79

Таблица 2 Поиск ключевых особенностей на изображениях «ROOFS»

Метод	Количество точек		Время поиска, мс
	Левая картинка	Правая картинка	
SIFT	3541	2416	117
ORB	9136	8246	470
BRISK	8046	5492	228
AKAZE	1996	1421	76

Таблица 3 Поиск ключевых особенностей на изображениях «RIVER»

Метод	Количество точек		Время поиска, мс
	Левая картинка	Правая картинка	
SIFT	8862	9346	359
ORB	10000	10000	525
BRISK	24049	24799	787
AKAZE	4176	4193	219

Таблица 4 Поиск ключевых особенностей на изображениях «НОМЕ»

Метод	Количество точек		Время поиска, мс
	Левая картинка	Правая картинка	
SIFT	2546	1317	558
ORB	9170	6773	511
BRISK	6551	2642	169
AKAZE	1789	693	386

Таблица 5 Поиск ключевых особенностей на изображениях «UTTOWER»

Метод	Количество точек		Время поиска, мс
	Левая картинка	Правая картинка	
SIFT	5314	4195	285
ORB	9839	9660	493
BRISK	8431	6724	242
AKAZE	2388	2237	159

Продолжение приложения А

Таблица 6 Сопоставление ключевых особенностей на изображениях «MOUNTAIN»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%
SIFT	211	12	14,2	199	4	13,4				198	32	13,3			
ORB							443	54	6,7				474	94	7,1
BRISK							218	11	6,3				244	38	6,4
AKAZE							86	1	20,1				86	8	20,1

Таблица 7 Сопоставление ключевых особенностей на изображениях «ROOFS»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%
SIFT	300	88	8,5	277	27	7,8				285	94	8,1			
ORB							139	136	1,5				149	115	1,6
BRISK							196	98	2,4				215	164	2,7
AKAZE							102	8	5,1				108	21	5,4

Таблица 8 Сопоставление ключевых особенностей на изображениях «RIVER»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%
SIFT	1180	857	13,3	1143	202	12,9				1162	294	13,1			
ORB							218	178	2,2				225	180	2,3
BRISK							1040	1180	4,3				1066	1717	4,4
AKAZE							379	46	9,1				382	90	9,2

Продолжение приложения А

Таблица 9 Сопоставление ключевых особенностей на изображениях «HOME»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%
SIFT	109	35	4,3	98	9	3,8				100	59	3,9			
ORB							63	110	0,7				71	137	0,8
BRISK							100	36	1,6				103	94	1,6
AKAZE							85	4	5,1				91	13	5,1

Таблица 10 Сопоставление ключевых особенностей на изображениях «UTTOWER»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%	совп.	время, мс	%
SIFT	754	227	14,2	724	58	13,6				738	157	13,9			
ORB							487	168	4,9				537	164	5,5
BRISK							827	118	9,8				855	328	10,1
AKAZE							379	16	15,9				383	38	16,0

Продолжение приложения А

Таблица 11 Оценка гомографии на изображениях «MOUNTAIN»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac
	совпадения														
SIFT	210	170	172	191	162	166				169	163	165			
ORB							423	378	400				394	401	462
BRISK							190	189	187				209	195	195
AKAZE							75	81	81				75	81	81
	% от сопоставления														
SIFT	99,5	80,6	81,5	96,0	81,4	83,4				85,4	82,3	83,3			
ORB							95,5	85,3	90,3				83,1	84,6	97,5
BRISK							87,2	86,7	85,8				93,3	87,1	87,1
AKAZE							87,2	94,2	94,2				87,2	94,2	94,2

Таблица 12 Оценка гомографии на изображениях «ROOFS»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac
	совпадения														
SIFT	206	211	204	199	184	142				201	194	143			
ORB							115	118	119				128	127	129
BRISK							140	144	89				173	154	113
AKAZE							84	80	82				87	84	90
	% от сопоставления														
SIFT	68,7	70,3	68,0	71,8	66,4	51,3				70,5	68,1	50,2			
ORB							82,7	84,9	85,6				85,9	85,2	86,6
BRISK							71,4	73,5	45,4				80,5	71,6	52,6
AKAZE							82,4	78,4	80,4				80,6	77,8	83,3

Продолжение приложения А

Таблица 13 Оценка гомографии на изображениях «RIVER»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac
	совпадения														
SIFT	1025	920	621	1077	950	960				1162	1105	473			
ORB							181	191	218				192	220	221
BRISK							988	841	638				787	802	646
AKAZE							378	318	347				351	329	348
	% от сопоставления														
SIFT	86,9	78,0	52,6	94,2	83,1	84,0				100,0	95,1	40,7			
ORB							83,0	87,6	100,0				85,3	97,8	98,2
BRISK							95,0	80,9	61,3				73,8	75,2	60,6
AKAZE							99,7	83,9	91,6				91,9	86,1	91,1

Таблица 14 Оценка гомографии на изображениях «HOME»

Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac
	совпадения														
SIFT	82	137	59	74	73	46				77	76	51			
ORB							53	60	60				67	68	68
BRISK							86	86	89				87	89	89
AKAZE							70	65	67				67	67	71
	% от сопоставления														
SIFT	75,2	68,8	54,1	75,5	74,5	46,9				77,0	76,0	51,0			
ORB							84,1	95,2	95,2				94,4	95,8	95,8
BRISK							86,0	86,0	89,0				84,5	86,4	86,4
AKAZE							82,4	76,5	78,8				73,6	73,6	78,0

Продолжение приложения А

Таблица 15 Оценка гомографии на изображениях «UTTOWER»

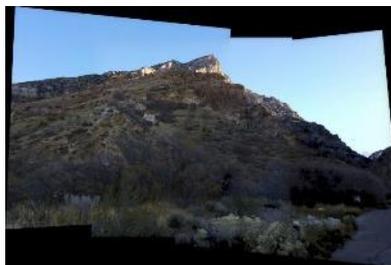
Метод	bf+norm_11			bf+norm_12			bf+norm_hamming			flann+kdtree			flann+ham(lsm)		
	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac	ransac	lmeds	prosac
	совпадения														
SIFT	707	723	701	723	720	708				737	711	733			
ORB							454	478	486				503	529	536
BRISK							822	807	518				853	827	679
AKAZE							371	371	374				353	376	377
	% от сопоставления														
SIFT	93,8	95,9	93,0	99,9	99,4	97,8				99,9	96,3	99,3			
ORB							93,2	98,2	99,8				93,7	98,5	99,8
BRISK							99,4	97,6	62,6				99,8	96,7	79,4
AKAZE							97,9	97,9	98,7				92,2	98,2	98,4

Приложение Б Результаты сопоставления с использованием вероятностных методов

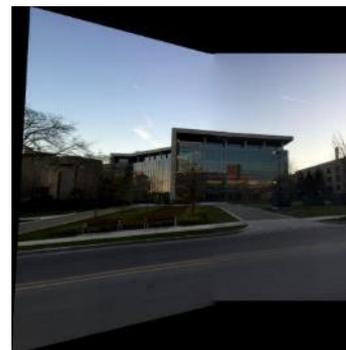
а) нет цветокорр. и смеш.

б) цветокорр. + альфа-смеш.

в) цветокорр. + многопол. смеш.



MOUNTAIN



HOUSE



RIVER



ROOFS



UTTOWER

Приложение В Результаты сшивки панорам в различных системах

Панорама набережной



БСИ



Hugin



Stitching API

Панорама улицы



БСИ

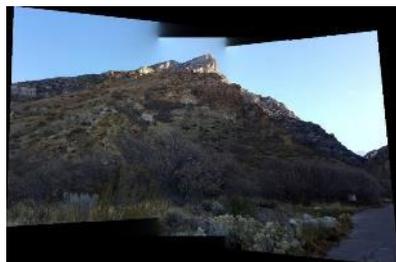


Hugin



Stitching API

Панорама MOUNTAIN



БСИ



Hugin



Stitching API

Панорама монумента



БСИ



Hugin



Stitching API

Панорама школы



БСИ

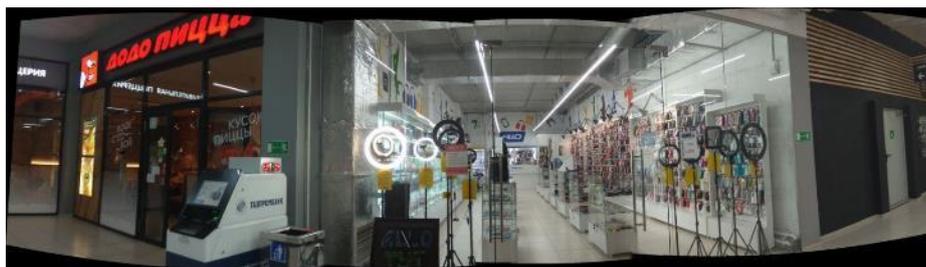


Hugin



Stitching API

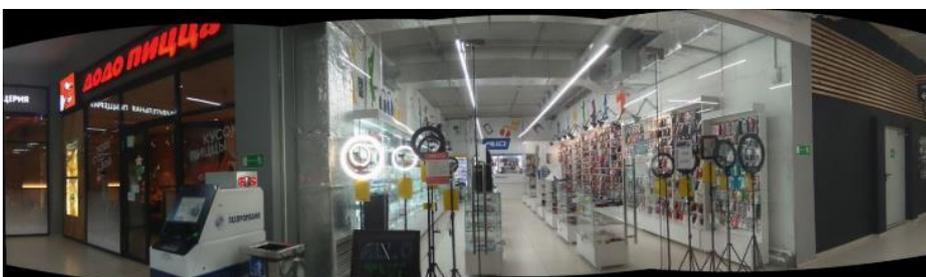
Панорама ТЦ



БСИ



Hugin



Stitching API

Панорама набережной



БСИ



Hugin



Stitching API

Приложение Г

Раздел 2

Development of algorithms for creating panoramic images

Студент:

Группа	ФИО	Подпись	Дата
8ВМ91	Унжакова Екатерина Викторовна		

Руководитель ВКР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Друки Алексей Алексеевич	к.т.н.		

Консультант – лингвист отделения ШБИП:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИЯ ШБИП	Аксёнова Наталья Валерьевна	к. филол. н.		

Introduction

Computer vision is a scientific direction in the field of artificial intelligence and related technologies for obtaining images of real-world objects, their processing and use of the data obtained to solve various types of applied tasks. All the tasks of computer vision are mainly reduced to the analysis of the image (s) or video stream (dialing of alternating images), which requires first allocate key features containing the necessary information.

Image registration is the process of comparing, combining and overlaying two or more images of the scene taken from different overview points. It is widely used in many apps for machine vision. Registration of the image consists of five main stages:

1. Feature Detection and Description.
2. Feature Matching.
3. Outlier Rejection.
4. Derivation of Transformation Function.
5. Image Reconstruction and Stitching.

Additional stage:

A developed software system for creating panoramic pictures from a series of overlapping scene frames is used approach based on the search for correspondences between images with the collection of statistical data and calculating promising cylindrical transformations.

The relevance of the development of the software system is due to the independence of the use of expensive video systems and the ability to obtain statistical data in the process of stitching images for research works.

1 Description of selected methods and means of development

1.1 Feature Detection and Description

To allocate key points, there are several basic methods already implemented in various software and widely used in practice.

To solve the problem of detection of special points on the images, a software implementation of a special algorithm in Python was implemented. As a means of organizing work with images, a computer vision library with open source - OpenCV (Open Source Computer Vision Library) was selected.

In this library there are various methods of computer vision and image processing, incl. SIFT (Scale-Invariant Feature Transform), ORB (Oriented Fast and Rotated Brief), Brisk (Binary Robust Invariant Scalable Keypoints) and AKAZE (Accelerated- KAZE). They are among the main signs of signs invariant with scalability, rotation and affine transformations. Algorithmically these are various methods that are in different image features.

The main point in the detection of special points by the SIFT method is to build the pyramid of Gaussians and the differences of Gaussians (DOG).

ORB uses as the basis a combination of the FAST detector and the Brief descriptor with a rotation of the calculation area in accordance with the orientation of a special point.

Brisk as well as ORB, created on the basis of a combination of the FAST detector with an optimized Brief descriptor, but changes have been made in their operation. To ensure the invariance of the method to scaling, the pyramid is also used, in which the search for singular points with the best maximum intensity in the pyramid using the FAST detector.

AKAZE is an advanced version of KAZE. As a descriptor, AKAZE uses M-LDB invariant to rotation and scaling.

These methods for finding key points are most popular and distributed (there are other, usually modifications specified). In addition, implementations are available in one open computer vision library OpenCV, which makes them irrelevant from

scratch (or the use of different sources / libraries) and makes it possible to compare their work in identical conditions. It is important.

1.2 Feature Matching

After detecting key features, the characteristics are compared with two mapping methods implemented via OpenCV: BFMatcher and FlannBasedMatcher.

BFMatcher accepts a descriptor of one object in the first set and is mapped to all other objects in the second set using some distance calculation, then the nearest one is returned.

FlannBasedMatcher contains a set of algorithms optimized to quickly find the nearest neighbor in large data sets and for multidimensional objects. It works faster than BFMatcher for large data sets. In other words, FLANN is much faster than BFMatcher, but he finds only an approximate nearest neighbor, which is a good coincidence, but not necessarily the best.

Figure 1 shows schematically metrics that assess the correspondence between the two methods under consideration by matching features in the OpenCV library.



Figure 1 Metrics for calculating the distance between key points in the methods BFMatcher and FlannBasedMatcher

BFMatcher uses the function of the L1-norm (Manhattan distance) or L2-norm (Euclidean distance) and Hamming distances.

Manhattan distance is medium differences in coordinates. In the distance of Euclidean, comparing each set of descriptors from the first image with each such set from the second image, we find the best match between special points. The Hamming distance is the number of distinguished bit in two binary vectors. The Hamming distance is already quite widely used for various tasks, such as searching for close

duplicates, image recognition, classification of documents, error correction, virus detection, etc.

FlannBasedMatcher uses an algorithm that uses a priority search to K-medium hierarchical trees (KDTree) and local hashing algorithm (LSH) to perform a search based on FLANN.

KDTREE (K-dimensional tree), a special data structure that allows you to break the K-dimensional space on «smaller parts», by section of this very space by hyperplanes ($K > 3$), planes ($K = 3$), straight ($K = 2$), and in the case of a one-dimensional space-point (performing a search in such a tree, it turns out something similar to the binary search). Such a partition is usually used to narrow the search range in the K-dimensional space.

LSH is a probabilistic method of lowering the dimension of multidimensional data. The basic idea consists of such a selection of hash functions for some measurements so that similar objects with a high degree of probability fall into one basket. The method allows you to build a structure for a quick approximate (probabilistic) search for n-dimensional vectors, «similar» to the desired pattern. LSH is probably the most popular today among the proposed approximate algorithms for finding the nearest neighbors.

Wealthy methods are different cluster methods based on the extent distance.

K-Means is a method of clustering (dividing the specified sample of objects (data) on non-passing subsets / groups (clusters) based on the proximity of their signs / values), striving to minimize the total quadratic deviation of cluster points from the centers of these clusters. K-Means is often given preference because of its simplicity of implementation and high speed.

1.3 Outlier Rejection

Incorrect coincidences (outliers) cannot be completely avoided at the stage of comparing the signs, therefore, for the exact fit of the conversion model, another phase of exclusion of unsuitable points is required. RANSAC, LMEDS and PROSAC are some of the reliable probabilistic methods used to remove incorrect coincidences from

the associated characteristics and selection of the conversion function available in the OpenCV library.

RANSAC

RANSAC is an iterative method for constructing reliable estimates of the parameters of a mathematical model based on a set of observed data, which are known to contain emissions. It gives a reasonable result only with a certain probability, and this probability increases with an increase in the number of iterations.

The RANSAC method can process almost any emission ratio, but it needs a threshold to distinguish emissions from emissions. RANSAC is often used in computer vision, for example, to simultaneously solve the problem of compliance and evaluate the fundamental matrix relating to the stereo chamber pair.

LMEDS

One of the earliest types of sustainable regression is the regression by the least squares method, which consists in minimizing the sum of the squares of the residues. In the particular case of simple regression, this corresponds to the search for the narrow strip that covers the half of observations, which reduces the effect of residues. According to Veneublubles and Ripley (1999), this algorithm minimizes the median value of ordered squares of residues to obtain the regression coefficient. For the LMEDS algorithm, a threshold value is not required, but it works correctly only if there are more than 50% of the labels. It is the easiest from a computational point of view. Estimates of regression coefficients at certain prerequisites have a number of optimal properties.

PROSAC

The structure of the PROSAC algorithm is similar to RANSAC. At first, hypotheses are generated by a random sample. Samples, unlike RANSAC, are not taken from all data, but from a subset of data with the highest quality. The size of a set for generating hypotheses is gradually increasing. Therefore, key features, which are more likely to with the least amount of noise, are investigated at an early stage. In fact, PROSAC is designed to draw the same characteristic features as RANSAC, only in a different order. Hypotheses are checked for compliance with all data. As in RANSAC,

the algorithm is completed when the probability of the existence of the solution, which at the moment it would be better than the best, becomes low (less than 5%).

1.4 Derivation of Transformation Function

Homography is a reversible transformation of a real projective plane on a projective plane that translates straight lines into straight lines. Homography is presented in the form of a 3×3 matrix with eight degrees of freedom.

The homographic matrix provides a promising transformation of the second image relative to the first (reference image). Then the image transformation is based on the resulting conversion function to align the second image relative to the first. The transformed version of the second image is then superimposed before the reference image.

The transformation of the perspective is the most complex form of two-dimensional plane transformations, which in order of increasing complexity include: translation, Euclidean transformation, conversion of similarity, affine conversion and projective transformation (figure 2).

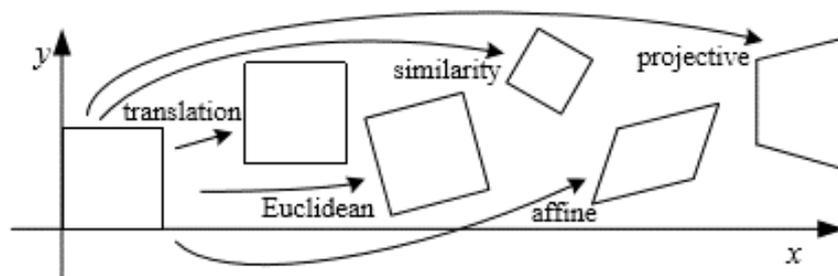


Figure 2 Basic set of 2D planar transformations

When the image area in which homography is calculated, small or the image was obtained with a large focal length, affine homography is a more suitable model stitching image.

In the graduation qualification work it was decided to implement the algorithm for a phased crossing of images with the definition of a homographic matrix and affine transformations.

1.5 Projections of panoramic images

After determining the homographic matrices of all input images, the choice of coordinate system, spaces for placing transformed images.

Starting work with panoramic images, you need to choose various existing panoramic projections: rectilinear projection, cylindrical projection, spherical (equidistant) projection, projection of the Mercator, cubic projection.

1. Rectilinear projection. Recognizing that the environment around you takes the form of the sphere, the rectilinear projection is used when only part of this sphere is photographed. Then this part of the sphere is projected onto a flat surface. It is often used to glue a small amount of images: well adapted for partial panoramas and mosaic images.

2. Cylindrical projection. It will be used when there is no data on Zenith or Nadir (sphere poles) on the panorama - it is usually a single-row image. This projection used for cylindrical panoramas and some larger partial panoramas, which look too distorted in a straight line projection.

3. Spherical (equidistant) projection. This projection is used to show the entire sphere projection on a flat surface. The result is an equally buggy image with the aspect ratio of 2: 1. This projection is used as a source for a complete spherical panorama.

4. Spherical (equidistant) projection. This projection is used to integrate the entire sphere on a flat surface. The result is an equally buggy image with the aspect ratio of 2:1. This projection is used as a source for full spherical panorama.

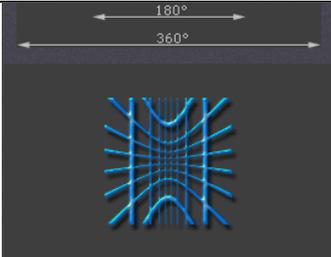
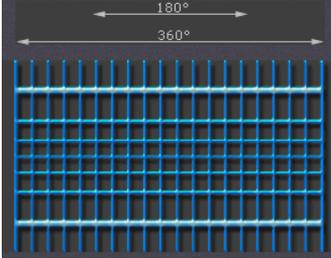
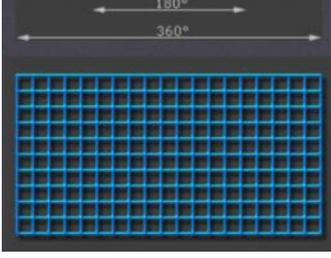
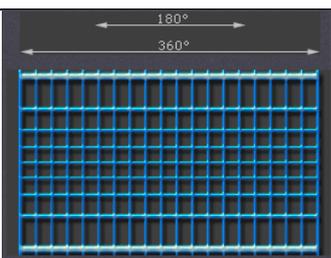
5. Projection of the Mercator. This projection can be considered as a derivative of an equally buggy projection without distortion in Zenith and Nadir. Maps of the world are in this projection. For panoramas, it may be an advantage if you need, for example, print an image of a spherical panorama or use the image on the website.

6. Cubic projection. This projection is used as a breakdown of an equally buggy image (spherical panorama). The principle is that the entire sphere turns into 6 faces of the cube - the front, right, rear, left, Zenit and Nadir. Each of the 6 square images has a straightforward projection.

This projection is very useful when you need to edit the image of a spherical panorama.

Each type of projection is trying to minimize one type of distortion due to the rest. Each projections have its advantages and disadvantages that are presented in Table 1.

Table 1. Types of projections of images, dignity and disadvantages

Type	Projection	Advantages	Disadvantages
Rectilinear		Can show full vertical and horizontal angles up to 360 degrees	Not suitable for images with a large vertical angle of view
Cylindrical		Retain more accurate relative sizes of objects	Not suitable for images with a large vertical angle of view
Equirectangular		Displays straight lines in three-dimensional space in straight lines on a flat two-dimensional grid	Significantly exaggerate the prospect of increasing the viewing angle, which leads to the visible objects in the logjam of the frame borders
Mercator		Closely correlated with cylindrical and equidistant projections; provides minimal vertical stretching and a wider angle of view than the cylindrical projection	Greater curvature of the horizontal lines

There are other projections, such as sinusoidal and stenographic, but when creating panorama, they are very rarely used.

1.6 Image Reconstruction

Often when crossing images in panorama on the final image you can see the effect of «mosaic», obtained as a result of changes in illumination and shades due to the different position of the chamber. Therefore, for a better perception of the panorama

resort to pre-correction color characteristics of the original images and methods of the visible «joint» correction between images.

Color and Brightness Correction

There are several image color correction algorithms. In a master's thesis, a method is considered based on the conversion of the RGB color model into the colors of CIELAB. After changes in the color channels occurred in the space, the colors are converted back to RGB. The space CIELAB is three-dimensional and covers the entire spectrum of human perception.

Thanks to the nature of the color definition in $L^* A^* B^*$, it is possible to separately affect the brightness, contrast of the image and on its color. In many cases, this allows you to speed up image processing. $L^* A^* B^*$ provides the ability to selectively influence individual colors in the image, strengthening color contrast, are irreplaceable and the possibilities that this color space provides to combat noise on digital photos.

Image Blending

1. Alpha Blending method

Very simple in part of the idea and implementation method. Based on the concepts of transparency of the image (alpha) and the blurring of the edge (Feathering). Figure 3 shows an example of how the effect of seamlessness is achieved due to the combination of these funds.

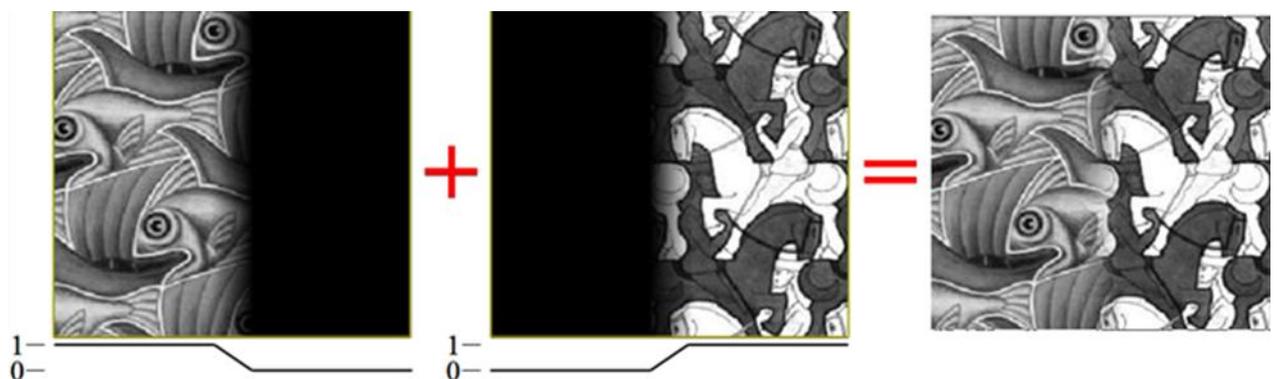


Figure 3 Alpha Blending

2. Method multiband blending using Gaussian and Laplacian pyramids

This method is more effective, although more costly in terms of performance than alpha mixing. It is built on the construction of the Gaussian

Pyramids and Laplace for images and the subsequent assembly of the outcome image.

Gaussian pyramid - a sequence of scaled versions of the same image (Figure 4).

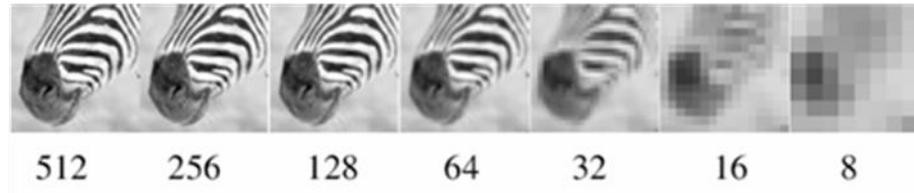


Figure 4 Gaussian pyramid

Laplace pyramid is a derivative of the Gaussian pyramid and has a similar structure. However, the image contained therein, in fact, processed by high-pass filter and thus only reflected parts and faces the image (Figure 5).

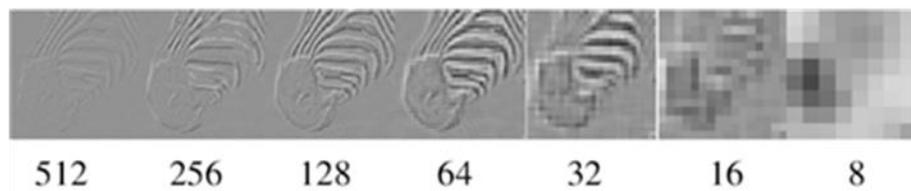


Figure 5 Laplacian pyramid

In order to blend two images, Laplace pyramids are built for both levels in size, the Gaussian pyramid (with the same level of levels) is also built for a special mask representing the mixing region.

Two pyramids are rejoined at each level using the appropriate levels of the mask pyramid. The result will be a mixed pyramid.

Thanks to the ability to restore a full-fledged image from the pyramid of Laplace and Gauss, a blending image is restored from the resulting pyramid, and the higher is blending quality, the greater the levels specify.

2 General Algorithm Software Software for Building Panoramic Images

2.1 Software Structure for Stitching Panoramic Images

The software system has been developed as a set of Python modules, each of which is specialized:

- «feature_detection.py» is a module for working with key points;
- «img_trim.py» is a small set of features for cropping images;
- «color_corrections.py» is a realization of color correction;
- «blendings.py» is a module for blending images;
- «panorama.py» is a module for creating panoramic images;
- «align_rotate_ends_blend.py» is a module for blending the edges have created panoramas.

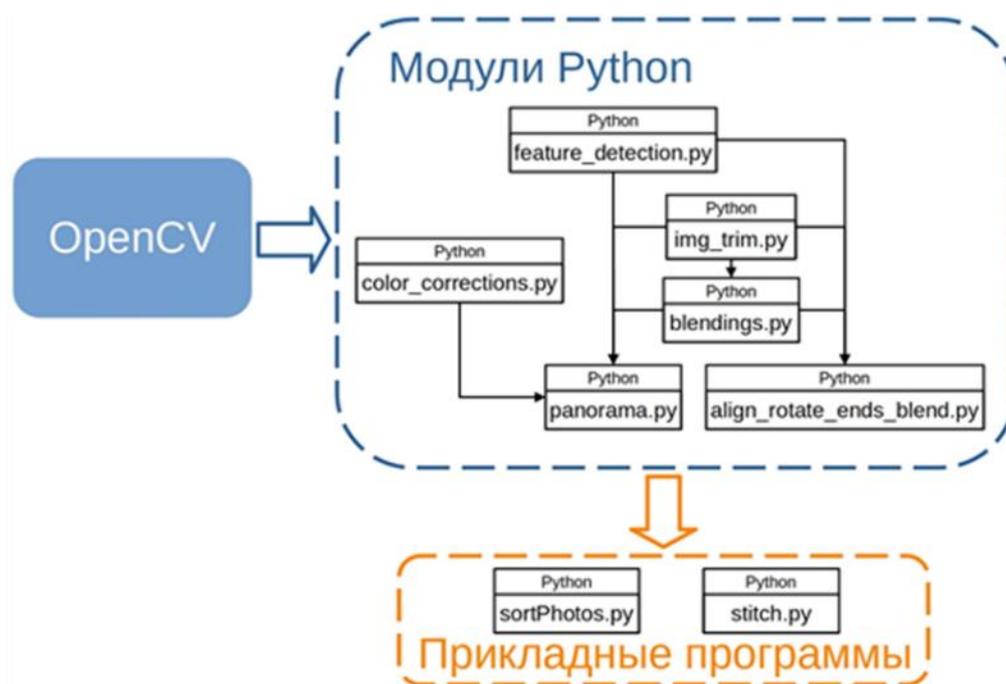


Figure 6 Modules and software tools

As a result, this set of modules is a small library based on OpenCV, suitable for use in applied, practical application, Python programs (Figure 6).

2.2 Visualization of panoramic image

To visualize panorama using web technologies, a JavaScript Panolens.js Library was used, which in turn requires the three.js graphics library. Demonstration

visualization deployed in the repository on the GitHub service using the Pages option, which allows the repository to be a hosting of static web pages [1].

Conclusion

As a result of working on the master's asset, several algorithms were proposed: sorting images, image blending, color correction, blending pair of images, cylindrical projection, creating a full-fledged panorama, as well as visualization of the finished panorama using web technologies.

As a result of the study, it can be concluded that the BRISK feature search method has the highest productivity - most of all key points found on test images and the smallest average search time. The best way to match for it is BFMatcher with the Hamming distance function and the probabilistic RANSAC method, since this combination gives the maximum number of points by which the crosslink will be made. This all suggests that this method can be quite effective for stitching frames into panoramic images, including in real time. At the same time, the results can vary on different parameters of images and cannot be excluded that other methods will be preferably preferred.

In turn, the probabilistic method of RANSAC has shown itself most effective in research conditions on a test sample of images.

The work in general, collected in its process of statistical data and their analysis, can be useful for decision-making in the process of developing algorithms and implementing computer-based applications in a wide variety of applications, such as virtual tours, aerial photography and mapping. That is, wherever the stitching of images can be involved and finding key features.