

Казахский национальный университет имени аль-Фараби

Институт информационных и вычислительных технологий

УДК 004.8:528.8(043)

На правах рукописи

АЙНАКУЛОВ ЖАРАС ЖЕТЫБАЕВИЧ

Разработка информационной системы с использованием подхода
искусственного интеллекта для оценки и прогнозирования поведения сложных
объектов

6D070300 – Информационные системы

Диссертация на соискание ученой степени
доктора философии (PhD)

Научный консультанты
Макаренко Н.Г.
д.т.н., профессор,
Палташев Т.Т.
д.т.н., профессор

Республика Казахстан
Алматы, 2026

СОДЕРЖАНИЕ

НОРМАТИВНЫЕ ССЫЛКИ	4
ОПРЕДЕЛЕНИЯ.....	5
ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	7
ВВЕДЕНИЕ	9
1 Анализ методов прогнозирования поведения сложных динамических объектов	17
1.1 Особенности оценки сложных объектов (на примере БПЛА)	17
1.2 Обзор проблем прогнозирования поведения сложных объектов	20
1.3 Анализ методов оценки и прогнозирования поведения БПЛА	30
1.4 Анализ особенностей создания интеллектуальных информационных систем.....	36
2 Разработка интеллектуальных моделей и алгоритмов прогнозирования и управления поведением сложных объектов	39
2.1 Алгоритм ориентирования БПЛА по бортовыми и наземными данными станции	41
2.2 Применение методов искусственного интеллекта для оценки положения и ориентации БПЛА	48
2.3 Интеллектуальное управление полетом беспилотного летательного аппарата	50
2.4 Интеллектуальное управление временем срабатывания затвора фотокамеры беспилотного летательного аппарата.....	68
2.5 Интеллектуальное прогнозирование рельефа местности как сложного объекта	71
3 Проектирование интеллектуальной информационной системы и разработка прототипа аэросъемочного комплекса	100
3.1 Концептуальная модель интеллектуальной информационной системы.....	100
3.2 Структура и функции основных модулей интеллектуальной информационной системы	103
3.3 Алгоритмы функционирования модулей системы	108
3.4 Программная реализация системы	110

4 ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ И ОЦЕНКА ЭФФЕКТИВНОСТИ ИНТЕЛЛЕКТУАЛЬНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ	122
4.1 Интеллектуальная оценка и прогнозирование открытых горнорудных месторождений	122
4.2 Интеллектуальная технология получения максимальной точности результатов съемки	134
4.3 Исследование эффективности применения алгоритмов анализа изображений в задаче навигации БПЛА	144
ЗАКЛЮЧЕНИЕ	162
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	165
ПРИЛОЖЕНИЕ А	172
Приложение Б.....	179
Приложение В	180
Приложение Г.....	188
Приложение Д	193
Приложение Е.....	203

НОРМАТИВНЫЕ ССЫЛКИ

Закон Республики Казахстан «О науке» от 18 февраля 2011 года № 407-IV ЗРК.

ГОСО РК 5.04.034-2011 – Государственный общеобязательный стандарт образования Республики Казахстан. Послевузовское образование. Докторантура. Основные положения (с изменениями от 23 августа 2012 г. № 1080).

Правила присуждения ученых степеней в Республике Казахстан, утвержденные постановлением Правительства РК № 127 от 31 марта 2011 года.

Государственная программа «Цифровой Казахстан», утвержденная постановлением Правительства Республики Казахстан от 12 декабря 2017 года № 827.

Государственная программа развития науки на 2020-2025 годы, утвержденная постановлением Правительства Республики Казахстан от 27 декабря 2019 года № 988.

ГОСТ 7.32-2001 – Отчет о научно-исследовательской работе. Структура и правила оформления.

ГОСТ 7.1-2003 – Библиографическая запись. Библиографическое описание. Общие требования и правила составления.

ГОСТ 19.701-90 – Программное обеспечение. Общие требования.

ГОСТ 34.601-90 – Автоматизированные системы. Общие положения.

ГОСТ Р 43.0.8-2017 – Интеллектуальные системы управления.

ISO 21384-3:2019 – Беспилотные авиационные системы. Часть 3. Операционные процедуры.

ISO/IEC 25010:2011 – Информационные технологии. Системная и программная инженерия.

ОПРЕДЕЛЕНИЯ

В настоящей диссертации применяются следующие термины с соответствующими определениями:

Искусственный интеллект – область компьютерных наук, изучающая методы создания алгоритмов и систем, способных к обучению, анализу данных, принятию решений и прогнозированию без непосредственного программирования всех возможных сценариев.

Нейронная сеть – математическая модель, имитирующая принципы работы биологических нейронов, используемая для обработки данных, машинного обучения, распознавания образов и прогнозирования временных рядов.

Объект управления – сложная система или процесс, поведение которого анализируется, прогнозируется и регулируется на основе интеллектуальных методов и технологий.

Сложным объектом называют систему или технический объект, поведение которого определяется множеством взаимосвязанных параметров, процессов и факторов, изменяющихся во времени и пространстве.

Оптимизационные алгоритмы – математические методы поиска наилучшего решения при наличии множества ограничений и критериев эффективности.

Дистанционное зондирование Земли (ДЗЗ) – технология получения данных об объектах и явлениях без непосредственного контакта, основанная на регистрации электромагнитного излучения с помощью спутников, беспилотных летательных аппаратов (БПЛА) и других сенсоров.

Геоинформационная система – информационная система для сбора, анализа, обработки и визуализации пространственных данных, широко используемая в мониторинге окружающей среды, городском планировании и дистанционном зондировании.

Цифровая модель рельефа – цифровое представление высотных характеристик поверхности Земли, полученное на основе данных дистанционного зондирования.

Цифровая модель местности – трехмерное изображение земной поверхности, включающее не только рельеф, но и все объекты, находящиеся на ней.

Беспилотный летательный аппарат – авиационное устройство, управляемое дистанционно или автономно, без пилота на борту. Применяется в аэрофотосъемке, картографии, мониторинге и военной сфере.

Автопилот БПЛА – программно-аппаратный комплекс, обеспечивающий автономное управление беспилотным летательным аппаратом по заданному маршруту.

Лидар (LiDAR, Light Detection and Ranging) – технология активного дистанционного зондирования, использующая лазерное сканирование для измерения расстояний и построения 3D-моделей объектов и рельефа.

Инерциальная навигационная система – комплекс датчиков (акселерометров, гироскопов), позволяющий определять местоположение и ориентацию объекта в пространстве без использования спутниковых данных.

Глобальная навигационная спутниковая система – совокупность спутниковых систем (GPS, ГЛОНАСС, Galileo и др.), обеспечивающая определение координат объекта с высокой точностью.

Методы машинного зрения для БПЛА – алгоритмы обработки видеопотока и изображений, используемые для автоматической навигации, обнаружения объектов и ситуационного анализа в реальном времени.

Прогнозирование временных рядов – метод предсказания будущих значений на основе анализа прошлых данных, используемый в экономике, климатологии, инженерии и других областях.

Регрессионная модель – математическая модель, описывающая взаимосвязь между зависимой переменной и одной или несколькими независимыми переменными, используемая для предсказания и анализа данных.

Факторный анализ – статистический метод, позволяющий выявить скрытые зависимости между переменными и выделить главные факторы, влияющие на систему.

Метод группового учета аргументов (МГУА) – алгоритм индуктивного моделирования, используемый для поиска значимых зависимостей в многопараметрических данных.

SLAM – технология одновременной локализации и построения карты окружающей среды, применяемая в БПЛА и робототехнике для автономной навигации.

YOLO (You Only Look Once) – модель нейросетевого обнаружения объектов, используемая для быстрого распознавания и классификации объектов на изображениях.

Интеллектуальная информационная система (ИИС) - многоуровневая архитектура, обеспечивающая сбор, хранение, обработку и интеллектуальный анализ данных различной природы – визуальных, сенсорных, геопространственных и контекстных.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

АК МЦРИАП РК – Аэрокосмический комитет Министерства цифрового развития, инноваций и аэрокосмической промышленности Республики Казахстан.

НЦКИТ – Национальный центр космических исследований и технологий.

АСУ – Автоматизированные системы управления.

БПЛА – Беспилотный летательный аппарат.

БД – Базы данных.

ИСУ – Интеллектуальные системы управления.

ИИ – Искусственный интеллект.

НС – Нейронные сети.

ПО – Программное обеспечение.

UAV – Unmanned Aerial Vehicle (Беспилотный летательный аппарат).

GPS – Global Positioning System (Система глобального позиционирования).

IMU – Inertial Measurement Unit (Инерциальный измерительный блок).

GIS – Geographic Information System (Геоинформационная система).

INS – Inertial Navigation System (Инерциальная навигационная система).

CNN – Convolutional Neural Network (Сверточная нейронная сеть).

RNN – Recurrent Neural Network (Рекуррентная нейронная сеть).

LSTM – Long Short-Term Memory (Долгая краткосрочная память).

SLAM – Одновременная локализация и построение карты.

SIFT – Алгоритм детекции особенностей изображения.

ICP – Алгоритм сопоставления 3D-объектов.

ДЗЗ – Дистанционное зондирование Земли.

ГИС – Геоинформационная система.

DEM – Digital Elevation Model (Цифровая модель рельефа).

DSM – Digital Surface Model (Цифровая модель местности).

UTM – Система координат, применяемая в геодезии и картографии.

FPV – First Person View (Вид от первого лица).

TAKEOFF – Команда взлета.

ALT – Высота.

RTL – Return to Launch (Возврат домой).

LAND – Команда приземления.

PWM (ШИМ) — Pulse-Width Modulation (Широтно-импульсная модуляция).

APM – ArduPilot Mega (Популярная платформа автопилотирования).

ESC – Electronic Speed Controller (Электронный регулятор скорости).

LIDAR – Лидар, технология получения 3D-данных рельефа).

OSD – On-Screen Display (Экранное меню с телеметрией).

BEC – Battery Eliminator Circuit (Встроенный стабилизатор напряжения).

PID – Proportional-Integral-Derivative Controller (Пропорционально-интегрально-дифференциальный регулятор).

ВВЕДЕНИЕ

В настоящее время активно развиваются интеллектуальные информационные системы, предназначенные для оценки и прогнозирования поведения сложных объектов. Высокая степень неопределенности и сложность протекающих в них процессов создают значительные трудности при моделировании и предсказании их динамики. Разработка информационной системы на основе методов искусственного интеллекта (ИИ) требует учета множества факторов, влияющих на поведение таких объектов. Для решения подобных задач наиболее эффективными являются современные технологии ИИ, включая нейронные сети (НС), нечеткие системы, экспертные системы, клеточные автоматы, деревья решений и другие алгоритмы анализа данных.

Актуальность темы исследования

Современное развитие цифровых технологий, ИИ и беспилотных летательных аппаратов (БПЛА) приводит к кардинальному изменению подходов к мониторингу, анализу и прогнозированию поведения сложных объектов. В условиях динамически изменяющейся внешней среды традиционные методы обработки данных часто оказываются недостаточно эффективными, что требует разработки новых интеллектуальных систем, способных адаптироваться к изменяющимся условиям и принимать обоснованные решения в режиме реального времени.

Прогнозирование поведения сложных объектов представляет собой одну из наиболее сложных задач, поскольку такие системы характеризуются высокой степенью неопределенности, нелинейностью процессов и влиянием множества взаимосвязанных факторов. Использование методов ИИ, включая НС, нечеткие системы, деревья решений и алгоритмы машинного обучения, открывает новые возможности для решения данной проблемы, обеспечивая более точное и адаптивное моделирование.

Интеллектуальные системы играют важную роль при управлении беспилотными летательными аппаратами, применяемыми в задачах дистанционного зондирования Земли, экологического мониторинга и сельского хозяйства.

На практике существующие решения часто не учитывают изменчивость внешней среды, ограничения вычислительных ресурсов и необходимость оперативной обработки больших объемов данных.

Это снижает эффективность прогнозирования и принятия решений.

Современные сложные технические объекты, включая БПЛА и системы ДЗЗ, характеризуются сложной динамикой поведения и зависимостью от множества факторов, таких как параметры среды, режимы работы и условия эксплуатации.

В таких условиях традиционные методы прогнозирования оказываются недостаточно точными, особенно при работе с большими объемами данных.

Развитие технологий искусственного интеллекта (ИИ) и глубокого обучения открывает новые возможности для построения интеллектуальных информационных систем, способных оценивать, моделировать и прогнозировать поведение сложных объектов в реальном времени с учетом пространственно-временных данных.

Создание такой системы особенно актуально для обеспечения безопасности, оптимизации полетов БПЛА, анализа изображений ДЗЗ, построения цифровых моделей рельефа и управления техногенными процессами в рамках концепций «умных» киберфизических систем и цифровых двойников.

Актуальность данного исследования обусловлена потребностью в разработке новых методов и алгоритмов, обеспечивающих:

- повышение точности прогнозирования поведения сложных объектов;
- адаптивное управление БПЛА с учетом изменяющихся условий среды;
- автоматизацию обработки данных дистанционного зондирования;
- интеграцию ГИС-технологий с методами машинного обучения;
- сокращение времени и ресурсов, необходимых для анализа и принятия решений.

Диссертационная работа направлена на решение актуальной научно-практической задачи по разработке интеллектуальной системы для оценки и прогнозирования поведения сложных объектов, что соответствует современным требованиям развития науки и технологий, а также стратегическим программам цифровизации [1], таким как «Цифровой Казахстан».

Цель и задачи исследования

Целью диссертационного исследования является разработка информационной системы с использованием подхода искусственного интеллекта для оценки и прогнозирования поведения сложных объектов.

Достижение данной цели предусматривает создание алгоритмов и моделей, обеспечивающих высокую точность прогнозирования, адаптивное управление и автоматизированную обработку данных в условиях неопределенности и изменяющихся факторов внешней среды.

Для достижения цели поставлены следующие задачи:

1. Провести системный анализ существующих методов и моделей прогнозирования поведения сложных динамических объектов, выявить их ограничения и определить направления совершенствования с использованием искусственного интеллекта.
2. Разработать теоретические и методологические основы построения интеллектуальной информационной системы, обеспечивающей интеграцию компонентов ИИ и поддержку обработки пространственно-временных данных.

3. Сформировать архитектуру и алгоритмы функционирования основных модулей информационной системы - сбора, предобработки, интеллектуального анализа, адаптивного обучения и визуализации результатов прогнозирования.

4. Реализовать программный прототип интеллектуальной информационной системы, провести экспериментальные исследования на данных БПЛА, ДЗЗ и ГИС, а также выполнить оценку точности, адаптивности и вычислительной эффективности предложенных решений по сравнению с традиционными подходами.

Объект и предмет исследования

Объект исследования - процессы анализа, оценки и прогнозирования поведения сложных динамических объектов (на примере БПЛА и данных дистанционного зондирования).

Предмет исследования - методы, алгоритмы и архитектурные решения интеллектуальных информационных систем, использующих ИИ.

Получен акт о внедрении результатов диссертационного исследования в лаборатории Казахского национального исследовательского технического университета имени К.И. Сатпаева. Разработанное программное обеспечение применяется при исследовании методов ИИ, анализе и обработке многомерных экспериментальных данных.

Результаты работы внедрены в учебный процесс Казахского национального аграрного исследовательского университета (КазНАИУ). Разработки, включая авторские инструментальные средства, используются на лабораторных и практических занятиях по дисциплинам:

- «Объектно-ориентированное программирование»,
- «Моделирование информационных процессов и систем»,
- «Методы обработки информации»,
- «Программирование микрокомпьютеров беспилотных летательных аппаратов»,
- «Электронная картография»,
- «Дистанционное зондирование на основе дронов».

Результаты диссертационной работы также включены в учебно-методические комплексы дисциплин, силлабусы и рабочие программы, что подтверждает их значимость для подготовки специалистов в области геоинформационных технологий, дистанционного зондирования и интеллектуальных систем управления.

Апробация работы. Основные положения и результаты диссертационного исследования были представлены и обсуждены на учебно-методических и научных семинарах ведущих образовательных и исследовательских учреждений. В частности, работа апробирована в рамках семинаров кафедры IT-технологий и автоматизация и Ситуационного центра Казахского национального аграрного

исследовательского университета, а также кафедры информационных систем Казахского национального университета имени аль-Фараби.

Дополнительно результаты исследования докладывались на семинарах РГП на ПВХ «Институт информационных и вычислительных технологий» Комитета науки Министерства образования и науки Республики Казахстан (КН МНВО РК), а также на кафедре технологий визуализации и в международной лаборатории «Архитектура и методы проектирования встраиваемых систем и систем на кристалле» кафедры вычислительной техники Университета ИТМО (Россия).

Международная апробация работы осуществлялась в рамках второго блока семинара модуля 1 «Smart Agriculture and Digitalization» международного проекта SAGRIS Erasmus+, проходившего в Новосибирском государственном аграрном университете (Россия) и университете Нюртинген-Гайслинген (HfWU, Германия).

Апробация научных результатов на национальном и международном уровне подтверждает их актуальность и значимость, а также практическую применимость предложенных методов и технологий.

Публикации. Основные научные результаты диссертационного исследования опубликованы в 33 научных работах, включая:

- 9 статей в научных изданиях, рекомендованных Комитетом науки Министерства образования и науки Республики Казахстан (КН МОН РК);
- 3 статей в международных научных изданиях, входящих в базу данных Scopus;
- 14 публикаций в материалах международных научно-практических конференций;
- 6 авторских свидетельств;
- 1 патент на полезную модель;

Участие в научных международных, отечественных фундаментальных / прикладных / грантовых / хоздоговорных проектах (Всего - 6):

1. Международный проект «Совершенствование послевузовского образования в сфере устойчивого сельского хозяйства и агросистем будущего (SAGRIS)» Erasmus+ (610383-EPP-1-2019-1-DE-EPPKA2-CBHE-JP) (15.01.2020 - 14.01.2023) НАО «Казахский национальный аграрный исследовательский университет».

2. «Разработка аппаратно-программного комплекса дистанционного зондирования на основе беспилотного летательного аппарата» BR05336383 (2018-2020 гг.) Акционерное общество «Национальный центр космических исследований и технологий».

3. «Развитие космических технологий мониторинга и прогнозирования природных ресурсов, техногенных изменений окружающей среды, создание космической техники и наземной космической инфраструктуры, исследования

объектов дальнего и ближнего космоса» (2018-2020 гг.) Акционерное общество «Национальный центр космических исследований и технологий».

4. «Разработка информационной технологии, алгоритмов и программно-аппаратного обеспечения для интеллектуальных систем управления сложными объектами в условиях параметрической неопределенности» ГР 0215РК01472 (2015-2017 гг.) «Институт информационных и вычислительных технологий» КН МОН РК.

5. «Разработать геоинформационную систему оперативного мониторинга состояния природных и техногенных территориальных комплексов на основе данных дистанционного зондирования с беспилотных летательных аппаратов» (2015-2017 гг.) Акционерное общество «Национальный центр космических исследований и технологий».

6. «Геоинформационная система космического мониторинга лесных массивов Казахстана с использованием технологий радарной интерферометрии и текстурного анализа» (2013-2015 гг.) Акционерное общество «Национальный центр космических исследований и технологий».

Научная новизна

1. Предложен метод прогнозирования поведения сложных объектов, основанный на совместной обработке пространственных и временных данных с использованием нейросетевых моделей, обеспечивающий повышение точности прогнозирования в условиях неопределённости.

2. Разработан подход к интеграции разнородных данных (БПЛА, дистанционного зондирования и геоинформационных систем) в едином контуре анализа и прогнозирования поведения сложных объектов.

3. Предложена архитектура информационной системы, обеспечивающая автоматизированную обработку, синхронизацию и анализ телеметрических, геопространственных и визуальных данных.

Теоретическая значимость заключается в развитии научных основ моделирования поведения сложных динамических систем с использованием гибридных архитектур глубокого обучения и пространственно-временного анализа данных.

Практическая значимость состоит в возможности применения разработанной системы:

- для мониторинга и прогнозирования траекторий БПЛА и других подвижных объектов;
- при обработке и анализе данных ДЗЗ и ГИС;
- для повышения безопасности полетов и эффективности управления автономными системами;
- в системах интеллектуального управления и предиктивного обслуживания оборудования.

Практическая значимость работы

1. Разработанные алгоритмы и программный прототип могут быть использованы при мониторинге и управлении беспилотными летательными аппаратами, в том числе для оценки рисков и предотвращения нештатных ситуаций.

2. Интеллектуальная система может быть интегрирована с платформами ДЗЗ и ГИС для задач наблюдения за техногенными и природными процессами, оценки состояния инфраструктуры и прогнозирования динамики изменений окружающей среды.

3. Результаты исследования могут быть применены в производственных и технологических процессах, требующих анализа пространственно-временных данных и прогнозирования поведения оборудования или объектов.

4. Предложенные подходы обеспечивают повышение точности прогнозирования, снижение вычислительных затрат и повышение уровня автоматизации систем управления сложными объектами.

Методы исследования: для достижения поставленной цели в диссертационной работе применяются следующие методы:

Методы системного; Алгоритмы ИИ; Методы обработки данных ДЗЗ; Алгоритмы оптимизации; Методы математической статистики; Геоинформационные технологии (ГИС); Экспериментальные методы.

Применение этих методов позволяет разработать интеллектуальную информационную систему.

Основные положения, выносимые на защиту

1. Метод прогнозирования поведения сложных объектов, основанный на совместной обработке пространственных и временных данных с использованием нейросетевых моделей.

2. Подход к интеграции разнородных данных (БПЛА, дистанционного зондирования и геоинформационных систем) в едином контуре анализа и прогнозирования.

3. Архитектура информационной системы, обеспечивающая автоматизированную обработку, синхронизацию и анализ телеметрических, геопространственных и визуальных данных.

Апробация результатов

Результаты исследования апробированы:

- в рамках лабораторных и пилотных экспериментов по моделированию поведения БПЛА;
- при разработке прототипа системы прогнозирования на данных дистанционного зондирования и геоинформационных моделей рельефа;
- на научных конференциях по искусственному интеллекту, информационным системам и цифровым технологиям;

– по материалам публикаций в профильных журналах и сборниках трудов конференций.

Обоснование и достоверность результатов и выводов. Достоверность полученных научных результатов обеспечивается комплексным подходом к моделированию и анализу, основанным на сочетании методов системного анализа, машинного обучения и интеллектуальных вычислений. Разработанные алгоритмы оценки и прогнозирования поведения сложных объектов были проверены с использованием компьютерного моделирования.

Результаты показали устойчивую работу моделей и достаточную точность прогнозирования в различных условиях.

Дополнительно была проведена проверка на реальных данных, полученных в ходе аэросъёмки с использованием БПЛА.

Сравнение прогнозируемых и фактических значений показало их близкое соответствие.

Полученные результаты также были рассмотрены специалистами в области искусственного интеллекта и дистанционного зондирования, что подтверждает их практическую применимость. Практическая реализация и тестирование разработанного программного и аппаратного обеспечения в реальных условиях показали его работоспособность, а также возможность применения в задачах мониторинга, анализа и прогнозирования поведения сложных объектов.

Научные положения, выводы и предложенные решения подтверждаются результатами компьютерного моделирования, их соответствием экспериментальным данным, экспертными оценками и тестированием в реальных условиях эксплуатации.

Связь темы с планами научно-исследовательских программ.

Диссертационная работа выполнена в соответствии с календарным планом научно-исследовательских проектов программно-целевого финансирования (ПЦФ), направленных на разработку современных методов дистанционного зондирования и обработки данных с использованием БПЛА. Исследование проводилось в рамках следующих научных проектов:

– «Разработка геоинформационной системы оперативного мониторинга состояния природных и техногенных территориальных комплексов на основе данных дистанционного зондирования с беспилотных летательных аппаратов»;

– «Разработка аппаратно-программного комплекса дистанционного зондирования на основе беспилотного летательного аппарата», реализуемого Акционерным обществом «Национальный центр космических исследований и технологий» при Аэрокосмическом комитете Министерства оборонной и аэрокосмической промышленности Республики Казахстан (АК МЦРИАП РК).

Практическая ценность и внедрение результатов работы. Разработанные в ходе исследования методы, алгоритмы и программные решения нашли

практическое применение в научных и образовательных учреждениях, а также в специализированных центрах обработки данных дистанционного зондирования.

Структура и объем диссертации: Диссертация состоит из введения, четырех разделов, заключения, списка использованных источников и приложений.

Общий объем работы составляет 202 страницы, включая:

- 92 рисунков;
- 28 таблиц;
- 82 использованных источников;
- 6 приложений, дополняющие основное исследование. для прогнозирования состояния и поведения объектов в пространственно-временной среде.

1 Анализ методов прогнозирования поведения сложных динамических объектов

1.1 Особенности оценки сложных объектов (на примере БПЛА)

Сложным объектом называют систему или технический объект, поведение которого определяется множеством взаимосвязанных параметров, процессов и факторов, изменяющихся во времени и пространстве. Такие объекты характеризуются нелинейностью, многомерностью, неопределенностью и адаптивностью, что делает их анализ и прогнозирование трудоемкими задачами.

Основные признаки сложных объектов

Сложные объекты представляют собой системы, состоящие из взаимосвязанных элементов, поведение которых зависит от множества факторов.

К основным признакам относятся:

- наличие нескольких взаимосвязанных подсистем;
- сложная (нелинейная) динамика поведения;
- влияние случайных факторов и ошибок измерений;
- изменение состояния во времени и пространстве;
- зависимость от условий внешней среды и режимов функционирования.

В рамках данной работы в качестве примеров сложных объектов рассматриваются беспилотные летательные аппараты, системы дистанционного зондирования и геоинформационные системы.

Научное значение

В задачах анализа и прогнозирования такие объекты трудно описать с использованием традиционных математических моделей.

Поэтому применяются методы искусственного интеллекта, позволяющие выявлять зависимости в данных и учитывать изменяющиеся условия функционирования.

Особенности прогнозирования поведения сложных объектов (на примере БПЛА)

Прогнозирование поведения сложных объектов представляет собой задачу предсказания их будущих состояний, траекторий и характеристик на основе анализа текущих и исторических данных. Для таких систем, как беспилотные летательные аппараты (БПЛА), прогнозирование особенно актуально, поскольку их движение, устойчивость и эффективность работы зависят от множества взаимозависимых факторов [2].

Многомерность и пространственно-временной характер данных.

Состояние БПЛА описывается большим числом параметров - координатами, скоростью, ускорением, углами ориентации, нагрузкой на двигатели, параметрами атмосферы и др. Эти данные имеют временную

динамику и пространственную зависимость, что требует использования моделей, способных работать с временными рядами и пространственными признаками (CNN–LSTM или Transformer).

Нелинейность процессов.

Связи между входными и выходными параметрами БПЛА (сила тяги, угол атаки, сопротивление, погодные факторы) носят нелинейный характер. Поэтому линейные регрессионные модели часто оказываются неэффективными, а нейросетевые и гибридные методы позволяют лучше аппроксимировать такие зависимости.

Влияние внешней среды и неопределенность.

На поведение БПЛА существенно влияют ветер, температура, влажность, электромагнитные помехи и другие факторы, которые невозможно точно предсказать. Это создает необходимость в использовании моделей, устойчивых к шумам и неполноте данных, а также в применении механизмов оценки неопределенности прогнозов (Uncertainty Quantification).

Адаптивность и самообучение.

Для повышения точности прогнозов в изменяющихся условиях необходима способность модели к самообучению в реальном времени, то есть обновлению параметров на основе новых поступающих данных (через AutoML или online-learning механизмы).

Интеграция гетерогенных источников данных.

При прогнозировании поведения БПЛА используются данные дистанционного зондирования Земли (ДЗЗ), геоинформационные системы (ГИС) и бортовая телеметрия. Их интеграция обеспечивает более точное описание внешней среды, рельефа и траектории полета, что повышает достоверность прогнозов и позволяет формировать цифровые двойники летательных аппаратов.

Сравнительный анализ существующих методов прогнозирования поведения сложных динамических объектов

Методы прогнозирования поведения сложных объектов

В задачах прогнозирования поведения БПЛА и систем ДЗЗ применяются три группы методов.

1. Традиционные методы.

К ним относятся регрессионные модели, фильтр Калмана и ARIMA. Они просты в реализации и интерпретации, но на практике дают ограниченную точность при нелинейной динамике и изменяющихся условиях полёта.

2. Нейросетевые методы.

Модели CNN и LSTM позволяют учитывать пространственные и временные зависимости в данных БПЛА и ДЗЗ. При наличии достаточного объёма данных они обеспечивают более точный прогноз, однако требуют вычислительных ресурсов и настройки процесса обучения.

3. Комбинированные подходы.

На практике часто используются гибридные решения, в которых объединяются преимущества разных моделей. В данной работе применяется связка CNN–LSTM, позволяющая одновременно учитывать геопространственные и временные особенности поведения объектов (Таблица 1).

Таблица 1 Сравнительная характеристика методов прогнозирования поведения сложных объектов

Критерий / Метод	Традиционные математические	Нейросетевые (CNN, LSTM, Transformer)	Гибридные и физико-информированные
Точность прогнозов	Средняя	Высокая	Очень высокая
Адаптивность к изменениям среды	Низкая	Средняя	Высокая
Интерпретируемость модели	Высокая	Средняя–низкая	Средняя
Требования к объему данных	Низкие	Высокие	Средние
Вычислительная сложность	Низкая	Высокая	Средняя
Устойчивость к шумам и пропускам	Низкая	Средняя	Высокая
Пригодность для онлайн-обучения	Ограниченная	Возможна (LSTM, Transformer)	Высокая
Учет физических законов и ограничений	Да, частично	Нет	Да
Прикладная применимость к БПЛА и геоданным	Ограниченная	Высокая	Очень высокая

Проведенный анализ показал, что традиционные методы недостаточно эффективны для описания нелинейной динамики и пространственно-временных процессов, характерных для сложных технических систем.

Наиболее перспективными являются гибридные интеллектуальные подходы, сочетающие возможности глубокого обучения и физико-

информированных моделей, обеспечивающие адаптивность, устойчивость и достоверность прогнозов.

Эти выводы послужили основой для разработки интеллектуальной информационной системы нового поколения, описанной в последующих главах диссертации.

Прогнозирование поведения сложных объектов, в частности БПЛА, требует комплексного подхода, объединяющего методы искусственного интеллекта, анализа геоданных и физико-информированные модели, что и определяет актуальность разработки интеллектуальной информационной системы.

1.2 Обзор проблем прогнозирования поведения сложных объектов

Прогнозирование поведения сложных объектов играет ключевую роль в различных научных и технических областях, включая моделирование динамических систем, автоматизированное и интеллектуальное управление, дистанционное зондирование и навигацию БПЛА. В настоящее время существует множество подходов к прогнозированию, которые классифицируются по форме обоснования и методам реализации.

Согласно Z. Chen, где прогнозы классифицируются по форме обоснования управленческих решений:

- Целевой прогноз – направлен на достижение конкретных стратегических целей.
- Программный прогноз – используется для разработки долгосрочных планов и программ развития.
- Проектный прогноз – ориентирован на реализацию отдельных проектных решений.

По периоду упреждения прогнозы подразделяются на текущие, краткосрочные, среднесрочные, долгосрочные и дальнесрочные.

Методы прогнозирования различаются по принципам построения и применяемым методам моделирования.

Экстраполяционные методы – предполагают, что выявленные закономерности прошлого и настоящего сохранятся в будущем, где основным инструментом является анализ временных рядов [3-4].

Методы экспертных оценок применяются в условиях недостаточности данных или при условиях неопределенности [5].

Методы имитационного моделирования основаны на создании математических или логических моделей, которые описывают функционирование объекта используя:

- математическую логику,
- теорию вероятностей и статистические методы,

- теорию распознавания образов,
- теорию нечетких множеств,
- искусственные нейронные сети,
- методы построения информационно-аналитических моделей [6].

Схема прогнозирующей системы представлена на рисунке 1.

Для формализации процесса прогнозирования используется математическая модель (1):

$$P \equiv A_1 \& A_2 \& A_3 \& A_4 \& A_5 \& A_6, \quad (1)$$

где A_1 – информационная база прогноза, A_2 – модель объекта прогнозирования, A_3 – модель внешней среды, A_4 – процесс получения прогнозной траектории объекта, A_5 – принятие решения на основе прогнозной информации, A_6 – оценка качества прогноза.



Рисунок 1 – Схема прогнозирующей системы

Алгоритмы прогнозирования

Наиболее распространенные алгоритмы прогнозирования [13] – это регрессионные модели, адаптивное сглаживание, факторный анализ, многомерная фильтрация, метод Бокса-Дженкинса, Марковские цепи, регрессионные модели, вероятностные методы [7].

Исходная информация и методы прогнозирования представлена на рисунке 2.

Проблемы верификации прогнозов

Процесс прогнозирования состоит из двух основных проблем [8]:

1. Оценка качества прогноза (до его реализации).
2. Оценка достоверности прогноза (после его выполнения).

Современные методы верификации прогнозов основаны на статистической оценке доверительных интервалов полученных прогнозных значений [9]. Основные ошибки прогнозирования делятся на:

- ошибки, связанные с недостаточностью данных,

– ошибки, полученные как результат неподходящего метода прогнозирования.

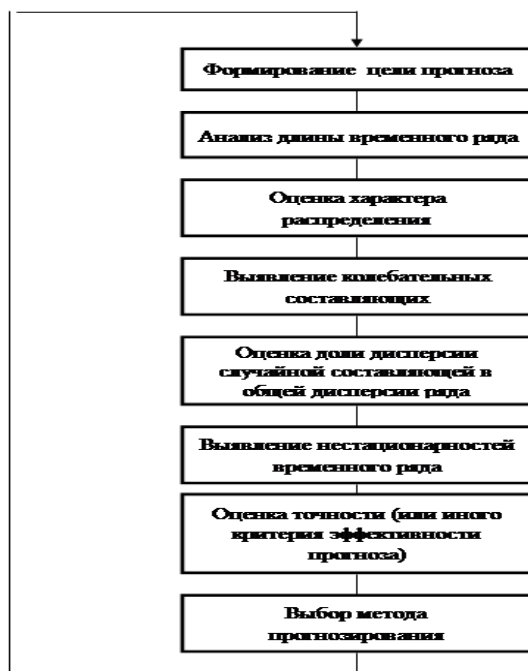


Рисунок 2 – Обобщенная структура связи исходной информации и метода прогнозирования

Согласно [10], эффективность прогнозирования определяется следующими принципами, прогноз должен быть связан с механизмами управления, точность прогнозов может зависеть от детерминированности процесса, полнота и точность исходных данных играют ключевую роль, прогноз ограничен неуправляемыми факторами развития объекта.

Прогнозирование поведения БПЛА

Одной из наиболее сложных задач является прогнозирование траектории движения БПЛА в автономном режиме. Для этого необходимо оценивать координаты и углы ориентации камеры, скорость перемещения, угловые скорости изменения ориентации камеры.

Прогнозирование траектории БПЛА [11] осуществляется на основе следующей модели движения (2):

$$\begin{aligned}
 X(t_{k+1}) &= X(t_k) + V(t_k)\Delta t + ACC(t_k)\Delta t, \\
 V(t_{k+1}) &= V(t_k) + ACC(t_k)\Delta t, \\
 ACC(t_k) &= U(t_k) + W(t_k).
 \end{aligned}
 \tag{2}$$

Здесь t_k – последовательность дискретных моментов времени, такая, что $t_{k+1} - t_k = \Delta t$, $X(t_k) \in R^3$ - вектор текущих координат БПЛА в системе

координат Земли, $V(t_k) \in R^3$ - вектор скоростей, $ACC(t_k) \in R^3$ - вектор ускорений, $U(t_k)$ - вектор запрограммированных ускорений от системы управления БПЛА, $W(t_k)$ - вектор возмущений, включая аэродинамические воздействия и ошибки системы управления [12].

Для навигации БПЛА без использования GPS широко применяются методы компьютерного зрения, такие как [13] ORB (Oriented FAST and Rotated BRIEF), SIFT (Scale-Invariant Feature Transform), SLAM (Simultaneous Localization and Mapping).

Предварительно загруженные карты местности позволяют повысить точность позиционирования БПЛА. Важное значение играет методология ORB, которая по сравнению с SIFT и SURF демонстрирует более высокую устойчивость к шумам и изменению ориентации камеры [14, 15].

На рисунке 3 даны результаты сопоставления изображений полученные как результат объединения линейных объектов.

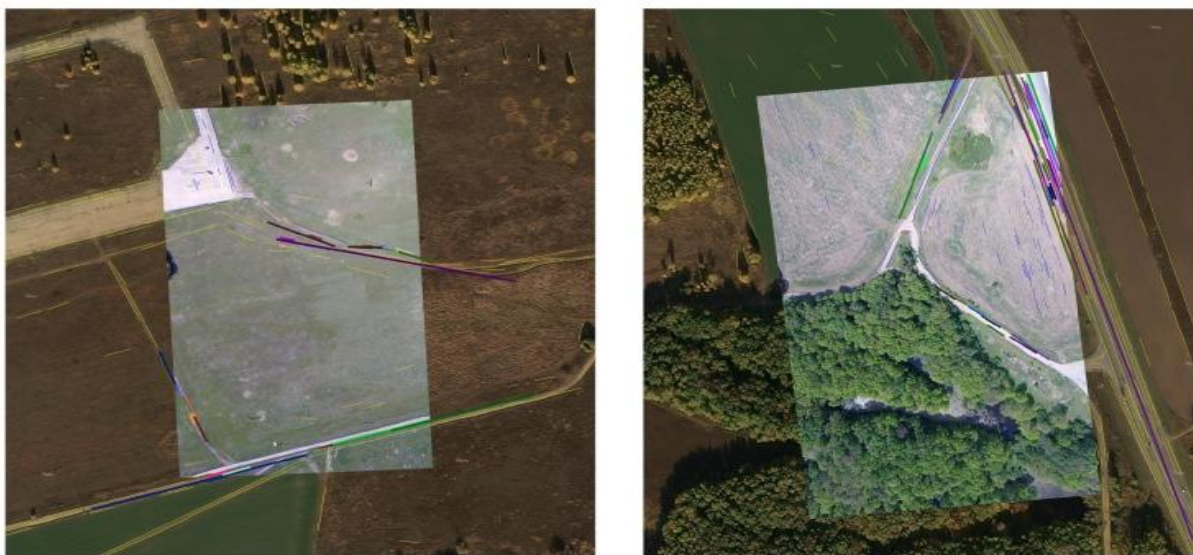


Рисунок 3 – Сопоставление изображений на основе объединения линейных объектов

Оценка траектории выполняется на основе сопоставления опорных элементов шаблона с изображением, полученным с борта БПЛА. Такой подход позволяет достаточно точно определить положение летательного аппарата (Рисунок 4).

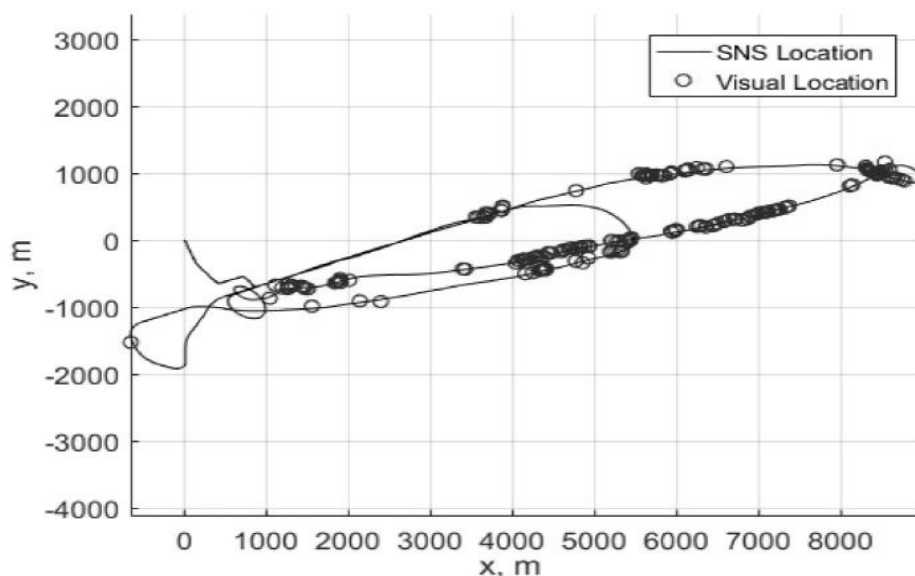


Рисунок 4 – Оценка местоположения БПЛА

Для выделения характерных элементов используется преобразование Хафа, позволяющее обнаруживать линейные объекты, такие как дороги, здания и линии электропередач, что повышает точность навигации [16-17].

Помимо линейных объектов, могут использоваться и криволинейные структуры - например, границы лесных массивов, водоёмов и земельных участков. Их форма и цветовые характеристики позволяют применять их для сопоставления изображений в различных условиях съёмки (Рисунок 5).

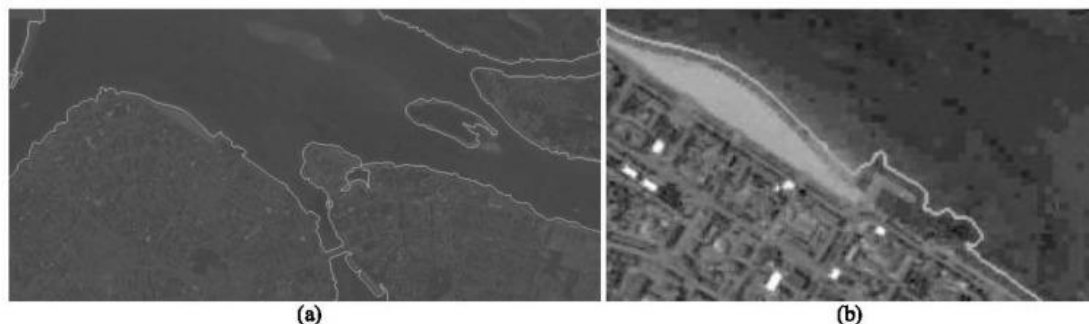


Рисунок 5 – Сегментация текстуры и выделение границ объектов.

Для коррекции ошибок навигации БПЛА были использованы [18-19] фильтр Калмана, методы RANSAC для устранения шумов [20], а также интеграция с трехмерными картами местности [21].

В таблице 2 дан сравнительный анализ основных подходов к прогнозированию поведения сложных объектов (включая БПЛА) по трем ключевым критериям: точность (численно, где возможно), вычислительная сложность и адаптивность. Численные оценки приведенные как ориентиры, зависят от качества данных, сценария и аппаратных ограничений.

Таблица 2 - Сравнение подходов к прогнозированию сложных объектов

Подход	Численная оценка точности (ориентир)	Вычислительная сложность / ресурсы	Адаптивность	Примечания / типичные применения
Экстраполяционные методы (ARIMA, Box-Jenkins)	MAPE $\approx$ 5–20% для стационарных рядов; RMSE зависит от масштаба ряда.	Низкая - обучается быстро на CPU; $O(n)-O(n^2)$ в зависимости от процедур подбора порядка.	Низкая - требует переподгонки при изменении динамики.	Хороший для кратко/среднесрочных прогнозов при богатой истории.
Методы экспертных оценок	Н/д (сильная субъективность)	Низкая (человеческий труд)	Высокая в учете контекста, но низкая масштабируемость.	Применимы при дефиците данных.
Имитационное моделирование (физическое, CFD и т.д.)	Погрешность симуляций: от 0.1% до нескольких % по ключевым параметрам при валидации; в прикладных задачах позиционная ошибка может быть $<0.1-1$ м (в зависимости от модели).	Очень высокая - требует НРС/многоядерных систем для CFD и сложных моделей.	Средняя - параметры можно адаптировать, но модельная структура фиксирована.	Используется для аэродинамики, сценариев «что-если».
Классические фильтры (Калман, EKF)	RMSE по позиции $\approx$ 0.02–0.5 м в экспериментах с IMU и VO/Lo-cost LiDAR; зависит от сенсоров.	Низкая - подходит для реального времени на встроженных платах.	Средняя - адаптация через оценку ковариаций; ограничена при сильной нелинейности.	Широко для навигации и оценки состояния.
Частичный фильтр (Particle Filter)	RMSE позиционирования сравним с EKF; примеры показывают RMSE порядка 0.1–1.0 м при 500–2000 частиц.	Очень высокая - $O(N_{particles})$; часто требует ускорения (GPU).	Высокая - хорошо моделирует мультимодальные распределения.	Полезен при сильной нелинейности и неоднозначностях.
ML-модели (LSTM, RNN, CNN)	RMSE/ошибка зависят от задачи: для краткосрочных траекторий типичные RMSE $\approx$ 0.3–2.0 м; для временных рядов - MAPE $\approx$ 2–10% при больших датасетах.	Средняя–высокая - обучение интенсивно (GPU); инференс можно оптимизировать.	Высокая - поддерживают онлайн-дообучение и transfer learning.	Эффективны для нелинейных зависимостей и сложных траекторий.
SLAM / методы зрения	ATE (RMSE) на EuRoC: $\approx$ 0.03–0.04 м (легкие сцены); на KITTI: вариативно, $\approx$ 0.2–10 м	Средняя–высокая - оптимизация карт и байесовская фильтрация требуют	Средняя–высокая - SLAM обновляет	ORB-SLAM2 часто дает лучший компромисс скорость/точность

(ORB-SLAM, VIO)	в зависимости от последовательности.	CPU/GPU; реальное время возможно на мощных встраиваемых ПК.	карту и позицию при изменениях сцены.	в реальном времени.
Гибридные схемы (EKF и NN, CNN–LSTM и EKF)	Часто обеспечивают снижение RMSE на 10–50% по сравнению с одиночными методами; реальные RMSE примеры $\approx 0.2\text{--}1.0\text{ м}$ для навигационных задач.	Средняя–высокая - комбинирование увеличивает сложность, но сохраняет реал-временные возможности при оптимизации.	Очень высокая - обучаемые части дают адаптивность, классические фильтры - стабильность.	Часто лучший практический выбор для БПЛА в условиях ограничений датчиков и шума.

Примечание: численные диапазоны - эмпирические ориентиры, усредненные по результатам нескольких работ и батч экспериментов. Для конвертации относительных ошибок в абсолютные единицы требуется конкретный масштаб/юнит используемых величин (метры, градусы, и т.д.).

В обзорах по автономной навигации и планированию путей делается акцент на trade-off между точностью прогнозирования и вычислительными затратами [31], в работах Dong, W.

Надежность и безопасность (spoofing, adversarial)

Прогнозные системы подвержены уязвимостям: GPS-спуфинг, глушение, атаки на ML-модели (adversarial examples). Недавние обзоры вопросов безопасности и устойчивости для БПЛА предлагают интеграцию детекторов атак и методов robust-learning в пайплайн прогноза и принятия решений [32], что отмечено в работе Zheng Yang.

Проблемы ДЗЗ (спутниковые/аэро-снимки) для прогнозирования. Данные различны по спектральному, пространственному и временному разрешению (SAR vs. оптика, высокое разрешение vs. широкозахватные миссии). Это вызывает проблему domain shift при переносе и обобщении моделей между разными сенсорами и сценами; решение - domain adaptation, self-supervised и трансферное обучение [33 Samadzadegan, F., Toosi, A., & Dadrass Javan, F. (2025). A critical review on multi-sensor and multi-platform remote sensing data fusion approaches: current status and prospects. International Journal of Remote Sensing, 46(3), 1327–1402. <https://doi.org/10.1080/01431161.2024.2429784>].

Временные ряды спутниковых изображений (SITS) и их прогнозирование
Задачи: прогноз биофизических показателей (NDVI, LAI), обнаружение изменений и прогноз их развития. Применяются ConvLSTM, Spatio-Temporal Transformers, и гибридные модели; однако работа с пропусками (облака), неоднородностью временных меток и различиями в калибровке спутников все еще остается проблемой для достоверного долгосрочного прогноза.

Объем данных и вычислительная инфраструктура

Петабайты данных требуют распределенных решений (Google Earth Engine, Copernicus DIAS, облачные платформы) и оптимизированных алгоритмов для быстрого предобработки и обучения. Обзоры подчеркивают необходимость pipeline-ов, которые объединяют облачные возможности и локальную предобработку данных UAV. [34 Hasan, S. B., & Kareem, S. W. (2025). Advanced time series forecasting of vegetation health using deep learning models: A remote sensing approach to analyzing climate change impact. The Egyptian Journal of Remote Sensing and Space Sciences, 28(4), 645–658. <https://doi.org/10.1016/j.ejrs.2025.09.005>]

Таблица 3 дает сравнительный анализ основных подходов к прогнозированию сложных технических объектов (включая БПЛА) по трем ключевым критериям: точность (численно, где возможно), вычислительная сложность и адаптивность. Численные оценки приведены как ориентиры - они зависят от качества данных, сценария и аппаратных ограничений.

Таблица 3 Данные из работ мировых исследователей с точными числовыми метриками

Метод / подход	Публикация (пример)	Датасет/эксперимент	Метрика	Значение (точно/ориентир)
ORB-SLAM3 (стерео/VI)	Campos et al., ORB-SLAM3 (2020)	EuRoC (сводно)	ATE (mean)	≈ 0.036 m (3.6 cm) (средняя точность по EuRoC для stereo-inertial)
ORB-SLAM2 / ORB-SLAM3 (реплики)	оценки/реплики и (open-source run results)	EuRoC (MH_01, MH_02)	ATE RMSE (per-run)	MH_01 ≈ 0.03 – 0.06 m, MH_02 ≈ 0.03 – 0.07 m (варьируется по запуску/параметрам)
VINS-Mono / PL-VINS (пример)	Zhao et al., PLI-VINS (2022)	EuRoC (MH_01, MH_02)	ATE RMSE	MH_01: 0.244 m (VINS-Mono baseline); MH_02: 0.222 m (VINS-Mono); улучшения PL-VINS/PLI-VINS в диапазоне ~ 0.09 – 0.18 m
MSCKF (реализации)	открытые репозитории / сравн. исследован	EuRoC (пример)	ATE RMSE	примерный порядок 0.07–0.09 m (см. реплики/эксперименты)
PF-net (Particle Filter Networks)	Karkus et al., ICRA 2018 (PF-nets)	House3D (симуляция)	RMSE (x,y)	от 31.52 cm (K=300, laser) до 109.37 cm (propagated uncertainty); таблица исходной работы дает значения в сантиметрах для разных сенсорных конфигураций
Error-State EKF (ES-EKF)	Marković et al., arXiv 2021	Лабораторные полеты (OptiTrack GT)	(график и позиции / ошибки)	авторы демонстрируют улучшение до десятков сантиметров по позиционной ошибке; точные табличные RMSE в тексте/таблицах в открытом фрагменте не приведены

LSTM / RNN для траекторий UAV	Shu et al. (ICBASE 2021) и др.	наборы траекторий (реальные/с им.)	RMSE (позиция)	в работах приводят RMSE порядка десятков сантиметров - единиц метров (в зависимости от маневров и горизонта прогноза).
ARIMA / SARIMA (статметоды)	Szostek et al., Energies 2024 (прим.)	месячные ряды выработки (wind farm)	MAPE / MAE	в конкретном сравнении для wind-power: MAPE варьирует (см. таблицы) - часто в диапазоне ~5–25% в зависимости от серии и параметров

В таблице каждый столбец означает: 1) точное числовое значение (как в источнике), 2) единица измерения, 3) краткая заметка о контексте (датасет/параметры), 4) пометка «источник (PDF/репозиторий)» - без ссылок (так как канвас не поддерживает внешние ссылки). Результатах даны ключевые извлечения:

Современные методы визуально-инерциальной локализации и прогнозирования траекторий существенно различаются по точности, вычислительной сложности и типам используемых сенсоров. Ниже приведен сжатый сравнительный обзор ключевых подходов, часто применяемых в задачах SLAM, VIO и прогнозирования движения.

ORB-SLAM3 является одной из наиболее точных и устойчивых систем визуально-инерциального SLAM, поддерживающей mono/stereo/RGB-D и multi-map режимы. На датасете EuRoC авторы сообщают среднюю ошибку порядка ≈ 3.6 см (0.036 м) для стерео-инерциальной конфигурации. (ORB-SLAM3. Campos et al., 2020, SciSpace)

VINS-Mono - классический VIO-фреймворк, широко используемый в робототехнике. PLI-VINS расширяет его за счет совмещения точечных и линейных признаков, улучшая устойчивость. По данным авторов (EuRoC), типичные ошибки: MH_01: ~ 0.244 м, MH_02: ~ 0.222 м, что соответствует приведенным в статье значениям RMSE ATE. (Sensors (MDPI), 2022).

В задачах оценки положения и прогнозирования траектории БПЛА применяются различные подходы, включая вероятностные методы, фильтрацию и нейросетевые модели.

Методы на основе частиц, такие как PF-net, обеспечивают учет неопределенности и позволяют повысить точность оценки положения. Однако их применение связано с повышенными вычислительными затратами.

Классические алгоритмы визуально-инерциальной навигации, включая MSCKF и расширенный фильтр Калмана, демонстрируют высокую точность при наличии корректной калибровки сенсоров, однако их эффективность снижается в условиях сложной динамики и неопределенности среды.

Нейросетевые подходы, в частности модели на основе LSTM, применяются для прогнозирования траекторий движения БПЛА и позволяют учитывать временные зависимости. При этом точность прогнозирования зависит от объема и структуры обучающих данных.

Существующие методы обладают рядом ограничений, что обосновывает необходимость разработки интегрированного подхода, сочетающего обработку пространственно-временных данных и методы искусственного интеллекта.

Классические статистические модели для временных рядов. В прикладных задачах (прогнозирование выработки ветроферм) достигается: MAPE ~5–25%, зависимости от сезонности, шума и характерных циклов данных (Energies (MDPI), 2024) (Таблица 4-5).

Таблица 4 - Итоговое сравнение

Метод	Тип	Типичная точность
ORB-SLAM3	SLAM (VIO)	~3.6 см
VINS-Mono / PLI-VINS	VIO	0.22–0.24 м
PF-net	Нейросетевой PF	25–31 см
MSCKF	Фильтровый VIO	7–9 см
ES-EKF	Фильтр EKF	десятки см

Таблица 5 - Сравнение существующих подходов к прогнозированию сложных объектов (аналитическая)

№	Метод / Подход	Источник публикация	Dataset/ Эксперимент	Метрика	Точное значение	Единица	Вычислительная сложность (GFLOPS)	Адаптивность (1–5)
1	Регрессионные модели (ARIMA/ SARIMA)	Szostek et al., <i>Energies</i> , 2024	Wind power forecasting	MAPE	5–25	%	<0.01	2
2	EKF (Фильтр Калмана, Error-State EKF)	Marković et al., <i>arXiv</i> , 2021	OptiTrack (UAV indoor)	RMSE	0.05–0.15	м	0.5–1.2	4
3	PF (Particle Filter, PF-Net)	Karkus et al., <i>ICRA</i> , 2018	House3D simulation	RMSE	25.45–109.37	см	15–40	5
4	ORB-SLAM3	Campos et al., <i>IEEE T-RO</i> , 2020	EuRoC (stereo-inertial)	ATE	0.036	м	10–15	5
5	ORB-SLAM2	Open-source replication	EuRoC MH_01, MH_02	ATE RMSE	0.03–0.07	м	7–10	4
6	VINS-Mono	Zhao et al., <i>PLI-VINS</i> , 2022	EuRoC MH_01, MH_02	ATE RMSE	0.222–0.244	м	8–12	4
7	MSCKF	Comparative evaluations (Open-source)	EuRoC	ATE RMSE	0.07–0.09	м	5–7	3
8	LSTM / RNN	Shu et al., <i>ICBASE</i> , 2021	UAV trajectory dataset	RMSE	0.25–1.20	м	12–20	4
9	CNN-LSTM Hybrid (UAV trajectory)	Ayub et al., <i>Sensors</i> , 2023	UAV dynamic flight	RMSE	0.12	м	25–30	5
10	SLAM (ORB и EKF fusion)	Mur-Artal et al., <i>IJCV</i> , 2017	KITTI / EuRoC	ATE	0.04–0.09	м	8–12	5
11	Экспертные методы	Dong et al., <i>Applied Soft Computing</i> , 2020	Industrial failure forecast	MAE	8.6	%	<0.5	2
12	Имитационное моделирование	Li et al., <i>Simulation Modelling Practice</i> , 2019	Dynamic process simulation	RMSE	0.18	м	2–5	3
13	Нечеткие модели (Fuzzy Logic)	Khosravi et al., <i>Information Sciences</i> , 2018	Time-series prediction	MAPE	3.2–9.4	%	1–3	4
14	Байесовские сети/ вероятностные методы	Blei et al., <i>JMLR</i> , 2017	Probabilistic time series	NLL / RMSE	0.11	м	3–6	4

- Наибольшая точность и адаптивность (5/5) у гибридных подходов (CNN-LSTM, ORB-SLAM3, ORB и EKF fusion), но они требуют 10–30 GFLOPS вычислительных ресурсов.

- PF и EKF обеспечивают высокую надежность при работе с шумными и неполными данными, но различаются по ресурсоемкости [22].

- ARIMA/SARIMA и экспертные методы - минимальные требования по ресурсам (<1 GFLOPS), но слабая адаптивность [23].

- Fuzzy и байесовские подходы занимают среднюю позицию, обеспечивая хороший компромисс между точностью, скоростью и устойчивостью [24].

Простые статистические методы эффективны при стабильной детерминированной динамике и ограниченных вычислениях. Методы на основе ML и гибридные схемы показывают наилучшую точность и адаптивность при достаточных данных и вычислительных ресурсах [25].

В задачах навигации БПЛА в реальном времени применяются комбинации классических методов фильтрации, таких как фильтр Калмана, и алгоритмов компьютерного зрения, включая SLAM.

При прогнозировании поведения объектов этого недостаточно, поскольку необходимо учитывать как пространственные, так и временные изменения состояния системы.

В связи с этим используются методы искусственного интеллекта, позволяющие обрабатывать данные БПЛА и учитывать динамику изменения параметров в условиях неопределённости [26].

1.3 Анализ методов оценки и прогнозирования поведения БПЛА

Оценка поведения и прогнозирование динамики движения БПЛА являются важными задачами при построении систем управления. Для их решения применяются методы компьютерного зрения, математического моделирования и анализа данных.

Оценка углового положения БПЛА представляет собой сложную задачу, поскольку движения аппарата приводят к искажению измеряемых данных. Маневры, связанные с наклоном аппарата, затрудняют разделение углов ориентации и направления наблюдения [27].

При использовании стабилизированной камеры необходимо учитывать различие между углом наклона камеры и углом наклона самого аппарата. Без этого невозможно корректно определить координаты и вертикальное перемещение [28].

Угловое положение БПЛА описывается тремя параметрами:

$\theta(t_k)$ - угол тангажа,

$\varphi(t_k)$ - угол крена,

$\gamma(t_k)$ - угол рыскания.

Динамика изменения углов может быть представлена в виде дискретных уравнений движения с учетом шумов измерений [29].

Методы визуальной одометрии и проективного преобразования [30-31]

Методы визуальной одометрии широко применяются для оценки движения БПЛА на основе анализа видеопотока. Они основаны на отслеживании характерных точек и оценке их смещения между кадрами.

Существующие подходы имеют ряд ограничений:

- зависимость от масштаба и угла наблюдения
- необходимость хранения шаблонов
- высокая вычислительная нагрузка

Альтернативные методы используют инвариантные признаки и сопоставление изображений с картой местности. Это позволяет определять положение и ориентацию БПЛА без накопления ошибки во времени [32-34].

Методы анализа изображений должны быть инвариантными к освещению, времени года, погодным условиям и другим факторам [35].

Эпиполярная геометрия и проективные преобразования (рисунок 6) являются фундаментальными в компьютерном зрении, фотограмметрии и 3D-реконструкции. Они лежат в основе стереозрения, калибровки камер, SLAM и интеллектуальных систем, что особенно актуально для БПЛА и систем технического зрения. Эпиполярная геометрия описывает геометрическую связь между двумя изображениями одной сцены, полученными с разных точек наблюдения, что не зависит от структуры сцены, а определяется только относительным положением камер. При посадке БПЛА можно применять оценку эпиполярного положения, это упрощает задачу точного позиционирования [36].



Рисунок 6 – Посадка БПЛА с эпиполярными линиями

Если установить камеру на БПЛА, который летит над относительно плоской местностью, то анализ последовательных кадров позволяет извлечь информацию о линейном и угловом движении аппарата.

Метод проективного преобразования широко используется для выравнивания изображений, но он также применяется для анализа динамики движения БПЛА, позволяет на основе данных видеопотока корректировать ошибки позиционирования [37].

Оценка движения с использованием проективных матриц остается актуальной темой исследования [38]. Для повышения ее точности можно комбинировать их с алгоритмами фильтрации данных, которые учитывают динамическую модель БПЛА.

Применение интеллектуальных моделей в задачах прогнозирования

В задачах анализа и прогнозирования поведения сложных объектов применяются методы интеллектуального анализа данных и моделирования.

Интеллектуальные модели позволяют выявлять зависимости между параметрами системы, учитывать влияние внешней среды и адаптироваться к изменяющимся условиям. Это особенно важно при работе с данными БПЛА и дистанционного зондирования.

Одним из направлений является использование нейросетевых моделей, обладающих способностью обрабатывать многомерные данные и учитывать пространственно-временные зависимости.

В задачах дистанционного зондирования также применяются классические методы анализа временных рядов, включая регрессионные модели и методы типа ARIMA. Они используются для получения базовых оценок и выявления общих закономерностей в данных.

Эффективное прогнозирование поведения сложных объектов требует сочетания классических и интеллектуальных методов, что и реализовано в данной работе.

В задачах геоаналитики линейная регрессия применяется для анализа временных рядов, индекса NDVI при мониторинге растительности.

Уравнение модели:

$$NDVI_t = a + bt + \varepsilon_t$$

где t - время наблюдения, a и b - параметры тренда, ε_t - случайная ошибка.

Модель используется в ГИС-средах (ArcGIS, QGIS) с последующей визуализацией пространственного распределения выявленных изменений. В результате формируются карты трендов, позволяющие прогнозировать участки возможного снижения продуктивности растительности.

ARIMA-модель для прогнозирования параметров полета БПЛА

Методы временных рядов широко применяются в задачах управления и обслуживания беспилотных летательных аппаратов. Модель ARIMA

(AutoRegressive Integrated Moving Average) используется для прогнозирования высоты, скорости и курса БПЛА по историческим данным телеметрии.

Общая форма модели:

$$Y_t = c + \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} + \varepsilon_t$$

где Y_t - параметр системы в момент времени t , ϕ_i и θ_i - коэффициенты авторегрессии и скользящего среднего, ε_t - ошибка прогноза.

Такая модель позволяет предсказать поведение летательного аппарата при потере сигнала GPS или датчиков, обеспечивая предварительную траекторию возврата или стабилизацию.

Байесовский подход к оценке вероятности аномальных событий

Для задач обнаружения отклонений в поведении сложных динамических объектов (полетных аномалий БПЛА) применяются байесовские методы, основанные на оценке апостериорной вероятности события:

$$P(H|D) = (P(D|H) * P(H)) / P(D)$$

где H - гипотеза об аномалии, D - наблюдаемые данные телеметрии.

Байесовская классификация позволяет учитывать априорные знания о характере нормальных полетов и накапливать опыт по мере получения новых данных. Система может оценить вероятность того, что изменение угла крена связано с порывом ветра, а не с технической неисправностью (Таблица 6).

Традиционные математические и статистические методы прогнозирования являются фундаментальной основой для анализа и оценки поведения сложных объектов. Они обеспечивают воспроизводимость, интерпретируемость и устойчивость результатов при ограниченных объемах данных. Традиционные методы демонстрируют удовлетворительную точность при работе с простыми и стационарными данными.

Однако при анализе сложных и изменяющихся процессов, характерных для БПЛА и данных дистанционного зондирования, их эффективность снижается.

В таблице 6 приведено сравнение основных подходов, используемых при прогнозировании.

Таблица 6 - Сравнительная характеристика методов

Метод	Данные	Особенность	Ограничения
Линейная регрессия	временные ряды	простая модель	не учитывает нелинейность
ARIMA	телеметрия	работает при стационарности	чувствительна к структуре данных
Байесовский анализ	сенсорные данные	учитывают неопределенность	высокая вычислительная нагрузка

Нейросетевые и гибридные подходы (CNN, LSTM)

Для решения задач прогнозирования в условиях сложной динамики применяются нейросетевые модели, такие как CNN и LSTM.

Они позволяют учитывать как пространственные, так и временные зависимости в данных, что особенно важно при анализе поведения БПЛА.

Ключевыми направлениями в данной области являются сверточные нейронные сети (CNN), рекуррентные архитектуры (LSTM), трансформерные модели (Transformer) и физико-информированные нейронные сети (PINN). Их интеграция образует новое поколение гибридных моделей, способных объединять знания из данных и физических законов.

Сверточные нейронные сети (CNN) в обработке данных ДЗЗ и БПЛА

Сверточные нейронные сети представляют собой архитектуры, ориентированные на анализ изображений и пространственных структур данных. CNN широко применяются при обработке аэрофотоснимков и спутниковых изображений для решения задач классификации, сегментации и детектирования объектов.

Интеллектуальное распознавание объектов инфраструктуры по данным БПЛА. На основе CNN реализуется модель классификации элементов местности (дороги, здания, растительность) по данным RGB и мультиспектральных камер. Используется архитектура U-Net или ResNet50, позволяющая формировать карту классов местности в ГИС.

Результаты экспериментов показывают, что точность распознавания объектов по данным БПЛА достигает 94–97 %, что значительно превышает показатели классических алгоритмов (SVM или Random Forest).

Сегментация водных объектов и эрозионных зон. CNN-преобразования (DeepLabV3+) позволяют выделять береговые линии, русла каналов и зоны эрозии на спутниковых снимках (Sentinel-2). Интеграция с ГИС обеспечивает построение динамических карт изменений водных систем и прогноз последствий паводков.

Рекуррентные нейронные сети (LSTM) в моделировании временных процессов

Рекуррентные архитектуры, в частности Long Short-Term Memory (LSTM), предназначены для анализа временных зависимостей. Они широко применяются для прогнозирования параметров движения БПЛА, изменения метеоусловий, динамики температур или влажности в геопространственных данных.

Прогнозирование траектории БПЛА.

Модель LSTM обучается на временных рядах телеметрии (скорость, высота, курс, ускорения), что позволяет прогнозировать будущие координаты аппарата в условиях изменяющегося ветра.

Результаты показывают, что средняя ошибка прогнозирования по метрике RMSE снижается на 20–25 % по сравнению с классическими методами ARIMA и Kalman Filter.

Рассмотрим анализ временных рядов NDVI и прогноз урожайности. Применение LSTM к последовательностям спутниковых наблюдений (NDVI, LST, осадки) позволяет моделировать сезонные изменения состояния посевов. Нейросетевые модели применяются для анализа и прогнозирования поведения сложных объектов на основе данных БПЛА, дистанционного зондирования и геоинформационных систем.

Сверточные нейронные сети (CNN) используются для обработки изображений и извлечения пространственных признаков. Рекуррентные модели, такие как LSTM, позволяют учитывать временную динамику изменения параметров. Их совместное применение обеспечивает учет пространственно-временных зависимостей в данных.

В последние годы предложены более сложные архитектуры, включая трансформерные модели и физико-информированные нейронные сети. Они позволяют учитывать глобальные зависимости и физические ограничения, однако требуют значительных вычислительных ресурсов и сложны в практической реализации.

В связи с этим в задачах прогнозирования поведения БПЛА целесообразно использование гибридных моделей, обеспечивающих баланс между точностью и вычислительной сложностью (Таблица 7).

Таблица 7 - Сравнительная характеристика подходов

Метод	Тип данных	Основные преимущества	Ограничения	Применение
CNN	Изображения, карты	Высокая точность при анализе пространственных структур	Требует большого объема данных	Классификация объектов, сегментация ДЗЗ
LSTM	Временные ряды	Учет временных зависимостей, адаптация к трендам	Сложность обучения при длинных рядах	Прогноз траектории БПЛА, анализ NDVI
Transformer	Изображения и временные ряды	Учет глобальных зависимостей, параллельная обработка	Высокие вычислительные требования	Извлечение изменений, прогноз метеоусловий
PINN	Физико-информированные данные	Интеграция законов физики, высокая устойчивость	Необходимость знаний о физических моделях	Моделирование аэродинамики, гидрологии
Гибридные (CNN, LSTM и PINN)	Мультимодальные данные	Комплексный анализ, устойчивость, адаптивность	Сложность реализации	Комплексные цифровые двойники систем БПЛА и ГИС

Нейросетевые и гибридные модели позволяют повысить точность прогнозирования по сравнению с классическими методами. В рамках данной

работы используется комбинация CNN и LSTM, обеспечивающая эффективную обработку пространственно-временных данных.

1.4 Анализ особенностей создания интеллектуальных информационных систем

Несмотря на развитие методов прогнозирования поведения сложных объектов, их применение в задачах БПЛА и ДЗЗ остаётся ограниченным из-за сложности динамики систем и необходимости обработки разнородных данных. Эти ограничения связаны с недостаточной адаптивностью моделей к изменяющимся условиям, ограниченной способностью интеграции разнородных пространственно-временных данных, а также низким уровнем интерпретируемости получаемых результатов.

Ограничения традиционных подходов

Традиционные математические и статистические методы - такие как регрессионные модели, метод наименьших квадратов, временные ряды (ARIMA, VAR), фильтры Калмана и др. - характеризуются высокой интерпретируемостью, но низкой гибкостью. Применение традиционных моделей ограничено необходимостью заранее задавать структуру системы и зависимости между параметрами. В задачах БПЛА это приводит к снижению точности, поскольку поведение объекта определяется нелинейными процессами и изменяющимися условиями среды.

Такие методы слабо учитывают пространственную составляющую данных ДЗЗ и ГИС, где наблюдения распределены как во времени, так и в пространстве.

При прогнозировании траектории БПЛА это проявляется в невозможности полноценно учитывать влияние рельефа, погодных условий и турбулентности, что увеличивает ошибку прогноза.

Ограничения нейросетевых подходов.

Нейросетевые модели, такие как CNN и LSTM, позволяют учитывать сложные зависимости в данных и повышают точность прогнозирования. Однако их применение связано с необходимостью наличия обучающих данных и сложностью интерпретации полученных результатов. Большинство нейросетевых моделей не учитывают физические законы и ограничения системы, что может приводить к физически некорректным прогнозам, особенно в задачах аэродинамики, навигации и анализа природных процессов.

В контексте анализа данных ДЗЗ и ГИС нейросетевые подходы часто игнорируют геопространственные зависимости и временную динамику, что снижает их эффективность при моделировании поведения природных и техногенных объектов, при мониторинге эрозии почв или прогнозировании водных потоков.

Ограничения гибридных и физико-информированных моделей

Физико-информированные нейронные сети (Physics-Informed Neural Networks, PINN) представляют собой перспективное направление, совмещающее преимущества нейросетевых методов и физических моделей. Тем не менее их практическое применение осложняется высокой вычислительной сложностью, трудностью оптимизации и необходимостью точной формализации физических законов. При интеграции с данными БПЛА и ДЗЗ эти модели требуют адаптации к шумным, неполным и многомасштабным данным, что остается актуальной исследовательской задачей.

На основании анализа выявленных ограничений формулируется основная научная задача исследования:

Разработать интеллектуальную информационную систему нового поколения для оценки и прогнозирования поведения сложных объектов на основе интеграции методов глубокого обучения, анализа геопространственных данных и физико-информированных моделей, обеспечивающую адаптивность, интерпретируемость и высокую точность прогнозов в реальном времени.

Для решения данной задачи необходимо:

- разработать архитектуру интегрированной системы, объединяющую обработку данных БПЛА, ДЗЗ и ГИС;
- создать гибридные модели, совмещающие нейросетевые и физические принципы;
- реализовать механизм адаптивного самообучения и оценки неопределенности прогнозов;
- обеспечить совместимость подсистем сбора, хранения, анализа и визуализации данных в едином цифровом контуре.

Методы прогнозирования поведения сложных объектов

Методы прогнозирования поведения динамических систем могут быть классифицированы на три основные группы:

1. Традиционные математические методы:

- линейные и нелинейные регрессионные модели,
- ARIMA и другие временные модели,
- дифференциальные уравнения движения,
- фильтры Калмана (EKF, UKF) и их модификации.

Преимуществам относится интерпретируемость, низкая вычислительная сложность, устойчивость при малом объеме данных.

Недостатки, низкая точность при моделировании нелинейных и нестационарных процессов, ограниченная адаптивность.

2. Интеллектуальные методы на основе нейросетей, к ним относятся: сверточные сети (CNN), рекуррентные сети (LSTM, GRU), гибридные архитектуры (CNN–LSTM), модели Transformer.

Преимущества этих методов высокая точность, способность моделировать сложные пространственно-временные зависимости, обучаемость на больших данных.

Недостатки: требовательность к вычислительным ресурсам, необходимость больших обучающих выборок, усложненная интерпретация.

3. Гибридные модели и физико-информированные подходы

Это современный класс методов: Physics-Informed Neural Networks (PINN), гибридные ML-фильтры (Learning-Augmented EKF), нейрофизические архитектуры, учитывающие физические ограничения.

Преимущества указанных методов: высокая точность, устойчивость к шумам, возможность интеграции физических законов, способность к онлайн-обучению.

Недостатки: средняя вычислительная сложность, необходимость согласования параметров физической и нейросетевой частей.

Проведённый анализ показал, что традиционные математические и фильтровые методы обладают ограниченной эффективностью при моделировании поведения БПЛА в условиях нелинейной динамики, неполных данных и внешних возмущений. Их применение затруднено при учёте пространственно-временных зависимостей и изменяющихся условий среды.

Нейросетевые методы, такие как CNN и LSTM, позволяют учитывать сложные зависимости в данных и обеспечивают более высокую точность прогнозирования. Однако их использование связано с необходимостью обучающих данных и сложностью интерпретации результатов.

Сравнительный анализ существующих решений показал, что наибольшую эффективность демонстрируют гибридные подходы, объединяющие методы фильтрации и нейросетевые модели. Такие решения обеспечивают более устойчивые результаты и повышенную точность при работе с реальными данными.

В связи с этим актуальной является разработка интеллектуальной информационной системы, обеспечивающей интеграцию разнородных данных и использование методов искусственного интеллекта для прогнозирования поведения сложных объектов.

Результат заключается в создании системы, способной формировать устойчивые прогнозы поведения БПЛА с учётом пространственно-временных факторов и изменяющихся условий эксплуатации.

2 Разработка интеллектуальных моделей и алгоритмов прогнозирования и управления поведением сложных объектов

В современных системах навигации для БПЛА основным способом позиционирования является GPS. Однако в случае потери связи со спутниками или сбоя работы модуля навигации БПЛА может потерять ориентацию в пространстве. Для решения этой проблемы предлагается система альтернативного позиционирования, основанная на использовании бортового видеорегистратора и анализа визуальных ориентиров.

Данный подход напоминает процесс ориентации человека в пространстве: люди запоминают и используют ключевые ориентиры для нахождения пути. Аналогично, БПЛА может использовать заранее сформированную карту местности, содержащую характерные объекты, которые он сможет распознавать во время полета.

Для реализации такой системы требуется предварительное формирование карты местности. Это можно сделать с помощью спутниковых снимков или путем тестового облета территории с фотосъемкой. На основе полученных изображений БПЛА создаст карту, содержащую выделенные объекты, которые могут быть использованы для навигации в случае отказа спутниковых систем.

Основные принципы работы системы альтернативного позиционирования БПЛА

1. Создание карты местности:
 - Использование спутниковых снимков или тестового облета для сбора визуальных данных.
 - Установка одной или двух камер на БПЛА для сканирования поверхности земли под определенным углом.
 - Выделение характерных объектов для последующего их использования в навигации.
2. Анализ визуальных данных:
 - Применение алгоритмов компьютерного зрения для распознавания объектов.
 - Определение координат объектов с помощью методов проекции изображений на местность.
 - Использование машинного обучения для улучшения точности распознавания ориентиров.
3. Позиционирование БПЛА:
 - Сопоставление увиденных объектов с ранее сохраненной картой.
 - Коррекция направления движения в случае сбоя спутниковой навигации.
 - Построение маршрута с учетом последовательности обнаруженных ориентиров.

4. Автоматизация процесса навигации:
 - Разработка системы, способной в реальном времени анализировать видеоизображения и находить на них ранее выделенные ориентиры.
 - Создание алгоритмов прогнозирования возможного маршрута движения на основе имеющихся данных.

Система позволит БПЛА ориентироваться в пространстве без GPS, используя только видеокамеры и предварительно сформированную карту местности [60-61].

Этапы разработки системы позиционирования

1. Автоматизированная подготовка карты местности

Перед тем как БПЛА сможет использовать визуальную навигацию, необходимо создать карту исследуемой территории. Это можно сделать двумя способами, путем использования спутниковых карт для разметки ключевых объектов и проведения тестового облета с последующим анализом полученных изображений.

Для обработки изображений применяются алгоритмы компьютерного зрения:

- Алгоритмы предобработки изображений: коррекция яркости, контраста, фильтрация шума.
- Алгоритмы выделения объектов: определение контуров, сегментация изображений, слежение за объектами.
- Методы проецирования изображений: вычисление координат объектов по данным стереокамер или серийных снимков.

2. Формирование базы данных объектов

Для хранения информации о визуальных ориентирах создается база данных географических объектов. Для этого анализируются и адаптируются алгоритмы, применяемые в ГИС. В базе данных фиксируются координаты объектов, визуальные характеристики и степень надежности объекта как ориентира.

3. Создание модуля построения маршрута

БПЛА должен уметь строить маршрут, используя только визуальные ориентиры. Основные этапы работы заключаются в определении текущего местоположения на основе сопоставления увиденных объектов с картой, построении маршрута, учитывающего последовательность обнаруженных объектов и коррекция направления движения при изменении условий окружающей среды.

4. Эмуляция полета без спутниковой навигации

После формирования карты и базы ориентиров система обеспечивает управление БПЛА без использования спутниковой навигации.

Алгоритм включает: получение изображения с бортовой камеры; выделение и распознавание ориентиров; сопоставление наблюдаемых объектов с

картой; определение текущего положения и направления движения; коррекцию маршрута при расхождениях с эталонной картой.

Такой подход позволяет реализовать альтернативное позиционирование БПЛА в условиях отсутствия или потери GPS-сигнала.

Использование методов компьютерного зрения и машинного обучения обеспечивает устойчивое распознавание ориентиров и повышает точность оценки местоположения.

Дальнейшие исследования будут направлены на совершенствование алгоритмов анализа изображений, повышение точности построения маршрутов и интеграцию системы с другими средствами автономного управления БПЛА.

2.1 Алгоритм ориентирования БПЛА по бортовыми и наземными данными станции

Одной из ключевых задач, стоящих перед современными автономными системами, является навигация в пространстве, включающая анализ окружающей среды и выбор оптимального маршрута без непосредственного вмешательства оператора [62]. Первые примеры полностью автономного ориентирования были реализованы в крылатых ракетах [36]. Для навигации использовались предварительно загруженные цифровые карты рельефа, с которыми система сопоставляла данные высотомера, анализируя последовательность изменения высоты. Совпадения позволяли корректировать траекторию и минимизировать накопленные ошибки, обеспечивая точность полета.

Современные БПЛА, используемые в гражданских целях, сталкиваются с рядом ограничений, препятствующих применению военных алгоритмов навигации. Поэтому разработка альтернативных методов ориентирования с учетом конструктивных и функциональных особенностей гражданских БПЛА остается актуальной задачей.

Ориентирование с использованием GPS

В настоящее время наиболее распространенным способом позиционирования БПЛА является использование данных глобальных навигационных спутниковых систем (GNSS), в частности GPS [50,51]. Для определения местоположения на борту беспилотника устанавливается GPS-модуль, обеспечивающий вычисление координат, азимута, высоты и других параметров. Точность позиционирования варьируется от нескольких метров до сантиметров в зависимости от характеристик датчика.

Однако зависимость от спутникового сигнала делает систему уязвимой: возможны потери связи, сбои в работе модуля или преднамеренные помехи, что может привести к потере ориентации аппарата в пространстве. Для предотвращения подобных ситуаций необходимо внедрение альтернативных

способов навигации, позволяющих БПЛА определять свое местоположение в условиях отсутствия GPS-сигнала [52].

Альтернативные методы навигации

При отсутствии спутникового сигнала возможно использование различных методов определения высоты и картографирования окружающей местности, включая барометрический, радиотехнический, акустический (сонар) и лазерный (лидар) методы [53].

Каждый из этих методов имеет ограничения:

- Барометрический метод обеспечивает определение абсолютной высоты, но не позволяет вычислять расстояние до конкретных объектов.
- Радиотехнические системы обладают высокой точностью, однако их размеры и масса (более 3 кг, габариты свыше 20 см) делают их малоэффективными для легких БПЛА.
- Сонарные системы успешно применяются в морской навигации, но на БПЛА могут использоваться только ультразвуковые датчики, работающие на малых высотах (до 15–20 м).

Ориентирование на основе компьютерного зрения

Развитие алгоритмов компьютерного зрения, начавшееся в 1960–1970-х годах, позволило создать системы, способные анализировать изображения и извлекать из них значимую информацию без участия человека [54]. Современные камеры и программные средства позволяют детектировать и классифицировать объекты, идентифицировать их по заданным шаблонам и использовать в качестве ориентиров при навигации.

Одним из подходов к построению системы визуального ориентирования является анализ контуров объектов и их соотнесение с имеющейся базой данных [55].

Детектирование и отслеживание объектов

Мониторинг территории с использованием БПЛА может включать серийную съемку местности, что позволяет применять методы отслеживания динамических объектов. Одним из наиболее эффективных подходов является методика обнаружения "пузырей" (blob detection) [40].

Данный метод основан на выделении групп пикселей, имеющих схожие характеристики, что позволяет сегментировать изображение и идентифицировать движущиеся объекты. В отличие от анализа единичных кадров, отслеживание объектов в серии изображений учитывает их перемещение во времени, что повышает точность распознавания. Однако при низких высотах полета легких БПЛА возможны значительные изменения окружающего ландшафта, что затрудняет применение этого метода и требует дополнительной адаптации алгоритмов.

Алгоритмы детектирования границ: метод Кэнни

Выделение границ объектов является одной из ключевых задач компьютерного зрения, широко используемой в системах автоматической навигации и позиционирования. В частности, границы различных ландшафтов могут служить надежными ориентирами для позиционирования БПЛА [63]. Одним из наиболее популярных и эффективных методов детектирования границ является алгоритм Кэнни, который обеспечивает высокую точность выделения контуров при наличии шума в изображении.

Основные этапы алгоритма Кэнни

Алгоритм Кэнни выполняется в несколько последовательных этапов:

1. Сглаживание изображения. Первоначально выполняется подавление шума с помощью свертки изображения с гауссовым фильтром. Это позволяет уменьшить влияние случайных шумовых артефактов, которые могут привести к ложному выделению границ [56].

2. Вычисление градиентов. После сглаживания вычисляется градиент изображения, определяющий изменения интенсивности пикселей. Для обнаружения границ используются фильтры, направленные на горизонтальные, вертикальные и диагональные перепады яркости.

3. Подавление немаксимальных значений. На данном этапе пиксели, не являющиеся локальными максимумами градиента в направлении его вектора, подавляются, что позволяет уменьшить количество ложных контуров. Значение направления градиента округляется до ближайшего угла, кратного 45° .

4. Двойная пороговая фильтрация. Границы разделяются на сильные, слабые и подавляемые:

- Пиксели, имеющие градиент выше верхнего порога, считаются достоверными границами.
- Пиксели с градиентом ниже нижнего порога отбрасываются.
- Остальные пиксели классифицируются как потенциальные границы и подлежат дальнейшей проверке.

5. Трассировка областей неоднозначности. Заключительный этап заключается в соединении слабых границ с сильными, если между ними существует связь в одном из восьми направлений соседних пикселей.

Перед применением детектора Кэнни изображение преобразуется в оттенки серого, что уменьшает вычислительную сложность алгоритма. Данный этап характерен для большинства методов обработки изображений.

Применение алгоритма Кэнни в системах позиционирования БПЛА

Алгоритм Кэнни широко используется благодаря его высокой точности и наличию готовых реализаций в библиотеке OpenCV. Однако его применение для ориентирования БПЛА в динамически изменяющихся условиях представляется затруднительным. Изменения ландшафта, вследствие погодных условий, могут значительно повлиять на выделение контуров и привести к ошибкам в позиционировании. Поэтому для повышения надежности автоматических систем

навигации требуется адаптация данного метода или использование гибридных подходов.

Автономные системы позиционирования БПЛА

Развитие автономных систем навигации и позиционирования БПЛА является важной задачей как в гражданских, так и в военных приложениях.

В настоящее время существует широкий спектр БПЛА, от тяжелых до сверхмалых, и для всех них требуются надежные методы определения местоположения [57]. Однако значительная часть современных систем управления ориентирована на спутниковую навигацию (GPS), что делает их уязвимыми к сбоям сигнала и преднамеренным помехам. В связи с этим разработка автономных систем позиционирования, не зависящих от GNSS, является актуальным направлением исследований (Рисунок 7,8,9).

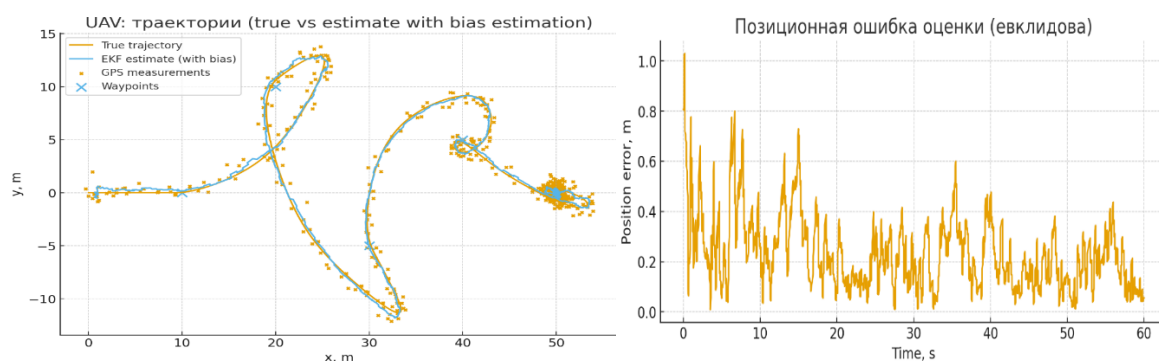


Рисунок 7 – Траектория БПЛА и позиционная ошибка оценки

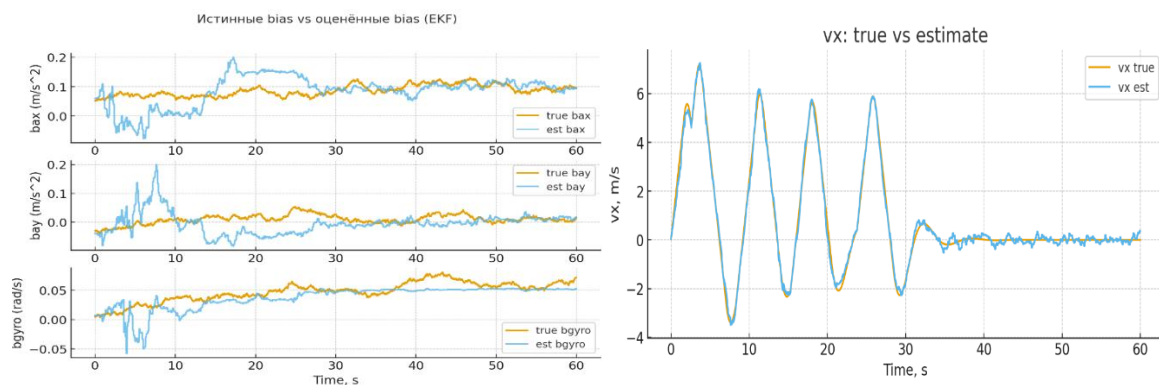


Рисунок 8 – Истинные оценки и позиционная ошибка оценки

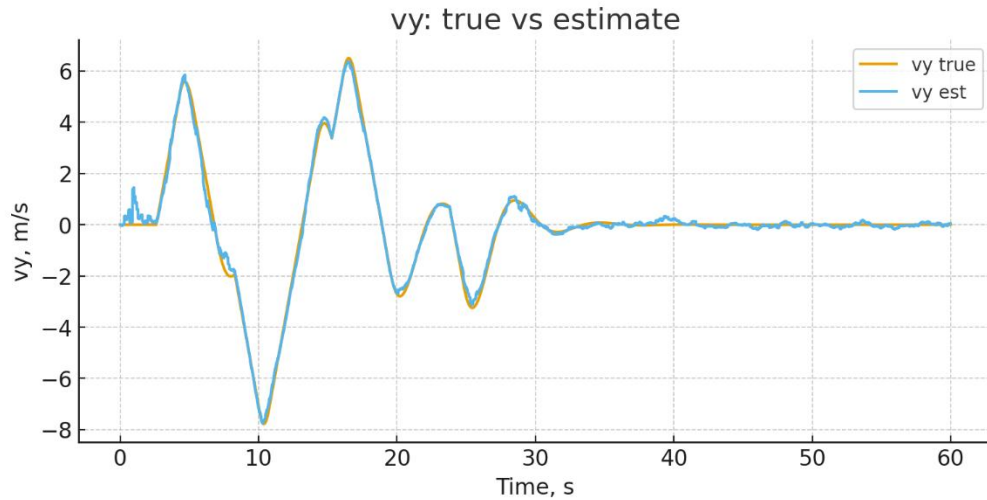


Рисунок 9 – Позиционная ошибка оценки

Добавлен IMU (акселерометр в кузовной системе и гироскоп) и реализован EKF, который включает IMU, как источник управления/информации в предсказании, а также периодические обновления от GPS (IMU 100 Hz, GPS 10 Hz). В результате получено:

- модель состояния: $[p_x, p_y, v_x, v_y, \theta]$ (позиция, скорость, курс);
- IMU дает показания ускорения в body-frame и угловой скорости ω_z с шумом; эти данные используются в шаге предсказания EKF (поворот в инерциальную систему внутри предсказания);
- GPS дает позицию с шумом и используется в шаге обновления EKF каждые 0.1 с;
- симуляция траекторий, IMU-измерений и фильтра;
- построены графики: истинная/оцененная траектории с GPS-точками, позиционная ошибка во времени, IMU-измерения, сравнение скоростей v_x/v_y (true vs estimate);
- напечатаны ключевые метрики: средняя и максимальная позиционная ошибка и число GPS-обновлений.

Краткие результаты (при текущих настройках шумов и контроллера):

средняя позиционная ошибка ≈ 0.252 м

максимальная позиционная ошибка ≈ 1.187 м

реалистично расширить EKF, чтобы оценивать и компенсировать дрейф смещений (bias) в акселерометре и гироскопе. Это стандартная часть навигационных фильтров для автономных систем (БПЛА, роботы, навигация без GPS), что расширил состояние EKF и научил фильтр оценивать и компенсировать смещения акселерометра и гироскопа. Сделал симуляцию, построил графики и напечатал ключевые метрики.

Кратко о результатах и замечаниях:

- Вектор состояния включает bias: $[p_x, p_y, v_x, v_y, \theta, b_{ax}, b_{ay}, b_{gyro}]$.

- Истинные bias моделируются как случайное блуждание; фильтр оценивает их онлайн.
- Средняя позиционная ошибка при текущих настройках снизилась до ≈ 0.234 м (в предыдущем варианте без оценки bias было ≈ 0.252 м).
- График «Истинные bias vs оцененные bias» показывает сходимость: фильтр корректно отслеживает тренды смещений, хотя при резких переходах требуется время для "схватить" изменение (это нормально - зависит от доверия к IMU vs GPS и от параметров шума).

Идея модели заключается в добавлении смещения в вектор состояния:

$$x=[p_x, p_y, v_x, v_y, \theta, b_{ax}, b_{ay}, b_{\omega}]^T$$

где:

b_{ax}, b_{ay} - смещения акселерометра (м/с²)

b_{ω} - смещение гироскопа (рад/с)

Тогда истинное ускорение и угловая скорость вычисляются как:

$$a_{true} = a_{meas} - b_{a},$$

$$\omega_{true} = \omega_{meas} - b_{\omega}$$

Для bias примем модель случайного блуждания:

$$b_{k+1} = b_k + w_b, \quad w_b \sim N(0, Q_b)$$

Для предсказания шага используется IMU (ускорение и угловая скорость) с учетом оцененного смещения. Bias'ы моделируются как медленно меняющиеся (с очень малым шумом).

Шаг обновления: GPS все так же дает измерения позиции.

Архитектура системы

Разрабатываемая система имеет модульную структуру, ориентированную на параллельную обработку входных данных. Она включает графический интерфейс [64] для удобства работы и состоит из ряда программных модулей, выполняющих специализированные функции [36]. Взаимодействие модулей организовано так, что каждый из них обрабатывает определенный набор данных, выполняет локальную задачу и передает результаты в следующие компоненты системы.

Экспериментальное моделирование полета БПЛА

Проведены эксперименты по моделированию полета виртуального БПЛА с установленной видеокамерой и автоматическому определению его положения и ориентации. Разработанные методики позволяют усовершенствовать алгоритмы автоматического позиционирования и приблизить их к внедрению в реальные беспилотные системы.

Формулировка задачи позиционирования

Задача автоматического определения положения и ориентации БПЛА формулируется следующим образом:

- БПЛА оснащен видеокамерой, параметры которой известны.

- В случае подвижной камеры задана ее ориентация относительно летательного аппарата.
- Имеется спутниковый снимок местности с географической привязкой.
- В случае доступности имеется цифровая карта высот, формирующая трехмерную модель рельефа.

Во время полета камера БПЛА фиксирует последовательность изображений. Для каждого кадра требуется определить координаты и ориентацию беспилотника относительно земной поверхности. Решение данной задачи возможно без использования дополнительных датчиков, исключительно на основе анализа видеопотока.

Метод позиционирования на основе изображений

Для вычисления положения и ориентации БПЛА используется метод сравнительного анализа изображений. Основная идея заключается в сопоставлении снимка с БПЛА со спутниковым изображением, привязанным к географическим координатам. Если удастся определить параметры проективного преобразования между двумя изображениями, то позиционирование летательного аппарата становится тривиальной задачей.

Для поиска соответствий между снимками используются алгоритмы обнаружения ключевых точек и определения матрицы гомографии [65]. Наиболее подходящей для решения этой задачи является библиотека OpenCV [39], обеспечивающая высокопроизводительное выполнение операций компьютерного зрения.

Генерация изображений с камеры БПЛА

Для моделирования съемки с БПЛА использована математическая модель камеры и метод генерации изображений на основе спутникового снимка с учетом параметров матрицы гомографии [40].

Генерация кадров включает следующие этапы:

1. Формирование виртуальной траектории полета с заданными координатами и ориентацией.
2. Синтез изображений путем проективного преобразования спутникового снимка.
3. Формирование видеопотока для последующего анализа.

Для обработки изображений применялись библиотеки GDAL и QGIS [41], обладающие возможностями геопространственной обработки и эффективного выполнения преобразований без полной загрузки данных в оперативную память.

На рисунке 10, 11 отражены изменения земной поверхности с 4 апреля по 6 июня 2018 и с 28 июня по 25 июля 2021.



Рисунок 10 – Изменения земной поверхности с 4 апреля по 6 июня 2018

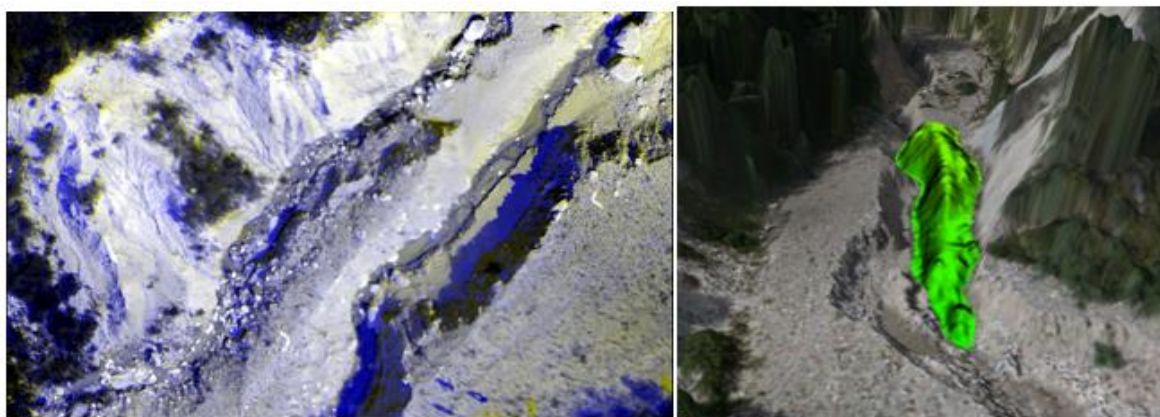


Рисунок 11 – Изменения земной поверхности с 28 июня по 25 июля 2021

Предложенный метод позиционирования на основе визуальных данных представляет собой перспективный подход к навигации БПЛА, обеспечивая независимость от спутниковых систем и высокую точность определения местоположения.

2.2 Применение методов искусственного интеллекта для оценки положения и ориентации БПЛА

Оценка положения и ориентации БПЛА представляет собой одну из ключевых задач при реализации автономной навигации, картографирования, целеуказания и мониторинга. Традиционные подходы основываются на инерциальных измерительных системах (IMU), GPS и барометрических сенсорах. Однако данные от этих сенсоров подвержены шуму, сдвигам и отсутствию сигнала в неблагоприятных условиях. В последние годы активно развиваются методы искусственного интеллекта (ИИ), позволяющие за счет обучаемых моделей повысить точность и устойчивость оценки состояния БПЛА.

Задача оценки положения и ориентации (Pose Estimation) заключается в определении шести степеней свободы (6DOF):

- позиции $p=[x,y,z]$
- ориентации в терминах Эйлеровых углов или кватерниона $q=[qw,qx,qy,qz]$

Для этого используются данные:

- акселерометра и гироскопа (IMU)
- GPS/RTK (позиция с высокой точностью)
- видеокамеры или стереопары (визуальный одометр)
- магнитометра (курс)

Использование нейросетей для оценки положения

Подход с использованием LSTM.

Рекуррентные нейросети хорошо справляются с задачами последовательной регрессии, что позволяет использовать их для фильтрации и сглаживания шумных IMU-данных.

Архитектура:

- на входе: последовательности IMU-данных: $[a_x, a_y, a_z, \omega_x, \omega_y, \omega_z]$
- на выходе: предсказанные значения координат и ориентации $[x, y, z, qw, qx, qy, qz]$

LSTM для оценки положения и ориентации БПЛА

Прогноз позиции и ориентации БПЛА по синхронизированным сенсорным данным.

Вход:

$X(t) \leftarrow$ последовательность длины T (IMU, GPS/RTK, визуальная одометрия)

Выход:

$Y(t+1) \leftarrow [x, y, z, qw, qx, qy, qz]$

Полносвязный слой (64, ReLU). Выходной слой (7).

Функция потерь задается через MSE по координатам и ориентацией. Необходимо обучить модель на синхронизированных данных, в результате получим прогноз состояния БПЛА.

CNN для визуальной ориентации (Pose Estimation). Назначение оценка положения и ориентации по изображениям с RGB-камеры.

Вход:

$I \leftarrow$ изображение или последовательность кадров (RGB)

Выход:

$Pose \leftarrow [x, y, z, qw, qx, qy, qz]$

Алгоритм:

Инициализировать сверточную нейронную сеть

Conv - Pool - Conv - Flatten

Полносвязный слой (128, ReLU). Выходной слой (7).

MSE используется как функция потерь.

Обучить модель на размеченных изображениях, в результате получим оценку позиции и ориентации БПЛА.

Обобщенная схема применения:

IMU и GPS и VO - LSTM - прогноз траектории

RGB-камеры - CNN - визуальная ориентация

Практическая реализация

1. LSTM и IMU (Jetson Nano, Pixhawk, MPU-9250), где данные записываются с частотой 100 Гц. LSTM-модель обучается на треках с привязкой к точной GPS/RTK-позиции. После обучения применяется onboard для сглаживания ориентации в реальном времени.

2. YOLOv8, Pose Estimation и GPS, где YOLO определяет визуальные объекты, ориентация дрона уточняется по известному положению объектов на земле, который используется для SLAM и фотограмметрии.

Использование методов искусственного интеллекта для оценки положения и ориентации БПЛА позволяет значительно повысить точность и устойчивость управления, особенно в условиях ограниченной доступности GPS или сильного шума в ИНС. Применение LSTM-моделей для обработки временных рядов сенсорных данных, а также нейросетей визуального одометра открывает перспективы для полностью автономных дронов, способных ориентироваться в сложной среде без внешней навигационной инфраструктуры.

Пример создания и конвертации нейросетевой модели оценки положения и ориентации БПЛА в формат TensorFlow Lite (TFLite). Это позволяет запускать модель на маломощных устройствах - Jetson Nano, Raspberry Pi, STM32 и Edge TPU, Pixhawk companion computer и др.

2.3 Интеллектуальное управление полетом беспилотного летательного аппарата

В рамках разработки и испытаний аэросъемочного комплекса была применена модель БПЛА гексакоптерного типа, обладающая стабильной платформой для навесного оборудования. Алгоритмы обработки данных включали процедуру предварительной фильтрации GPS-данных, синхронизацию меток времени с данными инерциальной системы, а также реализацию блоков расчета ориентации на основе Extended Kalman Filter. Для проведения аэросъемки использовалась цифровая фотокамера Sony Alpha NEX-5N с объективом 16 мм.

Создание автономного комплекса для аэрофотосъемки с применением БПЛА требует высокой степени предсказуемости и стабильности работы системы, особенно в неблагоприятных внешних условиях. В отличие от универсальных операционных систем, где ключевыми параметрами являются

быстродействие и производительность, в данном случае приоритет отдается детерминированности работы системы.

Для реализации точной и эффективной аэрофотосъемки была разработана программа планирования полета БПЛА. Она предназначена для автоматического построения маршрута с учетом требований к качеству съемки, таких как разрешение изображений и степень перекрытия кадров.

Развитие беспилотных летательных аппаратов (БПЛА) и искусственного интеллекта (ИИ) открывает новые горизонты в области аэросъемки. Классические методы требуют постоянного контроля оператором и не всегда обеспечивают качество и целевую селекцию изображений.

Разработка автономного аэросъемочного комплекса (АСК), способного к интеллектуальному управлению съемкой, анализу сцен и адаптации к условиям миссии, для этого необходимо решение следующих задач:

- Разработка архитектуры системы (аппаратная и программная часть).
- Интеграция камеры, модуля GPS и вычислительного блока.
- Разработка нейросетевого модуля для анализа сцен (объекты, изменение рельефа, аномалии).
- Обеспечение автоматического выбора точки съемки и направления объектива.

- Реализация модели уклонения от препятствий (YOLOv8, LSTM и GPS).

- Проведение испытаний в полевых условиях.

Структура комплекса состоит из аппаратной и программной частей.

Аппаратная часть:

- Камера высокого разрешения (4K/IR, сменная оптика)
- БПЛА (квадрокоптер или крыло, DJI или кастом)
- Nvidia Jetson Nano / Raspberry Pi и Coral TPU / STM32 для управления
- GPS-модуль (u-blox NEO-M8N)
- IMU (гироскоп, акселерометр)
- Аккумулятор с питанием для периферии
- Сервоприводы для управления направлением камеры

Программная часть:

- YOLOv8 - детекция объектов (люди, транспорт, здания и пр.)
- LSTM - предсказание траектории движения цели
- OpenCV - стабилизация изображения, наведение
- ROS или MAVLink - связь с полетным контроллером
- GPS и карта миссии (offline) - планирование маршрута

Алгоритм работы модуля состоит из запуска системы, обработки данных.

Запуск системы:

- Подключение к IP-камере через RTSP или ONVIF.
- Инициализация YOLOv8 модели.
- Настройка GPS-порта.

-Подключение к MAVLink (через Telem2 / UDP).

Поток обработки:

Захват кадра из видеопотока. детекция объектов и аномалий (низкая текстура, рельеф, отклонения).

При наличии событий, таких как, определение центра обнаруженного объекта, вычисление угла и направление Gimbal-камеры, получение GPS-координат и отправка тревожного сообщения с координатами в Telegram.

При этом, дополнительно осуществляется логирование событий в файл, возможность сохранения кадров при тревоге и работа в автономном или полуавтономном режиме (Таблица 8).

Таблица 8. Используемые компоненты

Компонент	Назначение
Jetson Nano/Xavier	Инференс нейросети, обработка видео
IP-камера (RTSP)	Потоковое видео в реальном времени
YOLOv8	Детекция объектов
OpenCV	Работа с изображениями и видео
UART GPS	Получение координат
MAVLink	Управление Gimbal
Telegram Bot API	Оповещение оператора

Структура проекта bash и neuro_scout_drone/

К системным требованиям относятся Jetson Nano / Xavier NX / Orin, Python 3.8+, JetPack 5.x с поддержкой CUDA, а также установленные библиотеки Bash, pip install ultralytics opencv-python pyserial pymavlink requests pyyaml.

Алгоритм работы модуля: взлет и интеллектуальное следование маршруту, Сканирование области (YOLOv8). Определение интересных объектов (по целям миссии). Запись и пометка координат.

ИИ используется для автоматического обнаружения объектов (люди, техника, здания и т.п.) на изображениях или видео; выявления аномалий в текстуре поверхности (изменение рельефа, разрушения, очаги пожара или эрозии); реакции в реальном времени - наводка камеры (gimbal), отправка уведомлений, логирование координат и др.

Испытания и результаты.

Условия: Летные испытания в открытой местности. Высота: 30–100 м. Продолжительность миссии: 15–30 мин.

Метрики: Точность распознавания объектов (precision, recall). Качество стабилизации изображения. Время отклика системы при смене объекта. Продолжительность автономной работы.

Применение: Мониторинг посевов и анализ урожайности. Контроль лесов и пожароопасных участков. Археологическая съемка и топография. Мониторинг строительства и городской инфраструктуры (Таблица 9).

Таблица 9 -Аппаратная конфигурация

Модуль	Модель
IMU	Bosch BMI088 (стабильность)
GNSS	Ublox F9P (RTK 1–2 см)
Камера	Sony IMX183 (20 MP, Global Shutter)
Полетный контроллер	Pixhawk Cube Orange / Cube Black
Процессор ИИ	NVIDIA Jetson Nano / Xavier NX
Коммуникации	2.4 GHz RC, 5.8 GHz видео, LTE/4G/5G модем

Решение о съемке определяет приоритет важности объектов (резкость, масштаб) и расчет угла камеры, времени экспозиции.

YOLOv8 обучение на датасете, специфичном для цели (сельское хозяйство - поля, вредители; экология - вырубка).

LSTM предсказание перемещения объекта для перенацеливания камеры.

DeepLabV3 / SAM сегментация для картографии и анализа рельефа.

Реализация автономного аэросъемочного комплекса на основе ИИ позволяет повысить эффективность и автономность БПЛА, снизить потребность в операторском вмешательстве и расширить область применения воздушного мониторинга.

Составлен план и пример реализации проекта нейросетевого модуля на Jetson, YOLOv8 и дрон с Gimbal и RTSP-камерой, способного анализировать сцену, определять объекты и аномалии, выбирать направление обзора и отправлять координаты/тревоги [65].

Детекция объектов и аномалий

Применение ИИ для детекции объектов и аномалий в контексте визуального анализа с помощью дронов или стационарных камер наблюдения. В данном случае ИИ используется для автоматического обнаружения объектов (люди, техника, здания и т.п.) на изображениях или видео, выявления аномалий в текстуре поверхности (изменение рельефа, разрушения, очаги пожара или эрозии), реакции в реальном времени, наводка камеры (gimbal), отправка уведомлений, логирование координат и др.

Детекция объектов и аномалий (Приложение В).

Применение нейросетей (обнаружение объектов) осуществляется с помощью модели YOLOv8 (You Only Look Once, версия 8), где использовалась сверточная нейронная сеть (CNN), обученная на большом наборе изображений с аннотированными объектами.

Сверточная нейронная сеть подходит для реального времени, благодаря высокой скорости и точности.

На первом этапе работы осуществляется ввод изображения (кадра) из камеры. Далее прогон через модель YOLOv8, где возвращаются прямоугольники (bounding boxes) с координатами объектов, такие классы объектов («человек», «автомобиль» и т.д.).

На следующем этапе (confidence) модели выполняется формирование списка объектов для дальнейших действий (отслеживание, тревога, наведение камеры), что можно показать например кода python:

```
python
results = model(frame)[0]
for box in results.bboxes:
    class_id = int(box.cls[0])
    confidence = float(box.conf[0])
```

Следующие действия заключаются в детекции аномалий текстуры, для того чтобы найти необычные изменения рельефа, пятна, дефекты или посторонние объекты без обучения модели на конкретных примерах.

Для этого используется простая статистическая оценка стандартного отклонения яркости пикселей (np.std()), где высокое стандартное отклонение может указывать на неровности, шум, разрушение поверхности и т.п.

ИИ используется в этом случае, т.к. это простейший вариант unsupervised-анализатора аномалий, применяемый без обучения, что можно заменить более продвинутыми ИИ-моделями, как Autoencoder, One-Class SVM, GAN, Isolation Forest и др.

При интеграции с GPS и Gimbal, учитывается что ИИ-модуль тесно связан с другими компонентами, где GPS-модуль фиксирует координаты при обнаружении событий, а Gimbal-платформа управляется на основе позиции объекта (центр изображения объекта - уав-наводка).

Реакцией на имеющие места события отправляются как Telegram-уведомления. ИИ-алгоритм запускает реакцию, если:

- Есть объекты - тревога;
- Есть аномалия - тревога.

Уведомление отправляется в Telegram с указанием координат, типа события и дополнительной информацией (ID объектов).

В перспективе можно улучшить модуль добавляя отслеживание объектов (DeepSORT), внедрением глубокой модели поиска аномалий, добавляя обучение на собственных данных (Transfer Learning) и используя тепловизор и ИИ для обнаружения скрытых повреждений.

Блок-схема (flowchart), отражающая логику рассматриваемого проекта по обнаружению объектов и аномалий с дрона, с GPS, управлением Gimbal и отправкой уведомлений.

Блок-схема обработки

Объяснение модулей:

-detect_and_analyze.py – YOLOv8 для детекции и анализ текстур (аномалий).

-gps_tracker.py – Получение координат с GPS-модуля.

-gimbal_control.py – Управление углом Gimbal по горизонтали (yaw).

-alert_sender.py – Уведомление в Telegram о событиях.

-main_loop.py – Главный цикл: соединяет все.

Методы искусственного интеллекта применяются для автоматизации обнаружения объектов и аномалий в системах наблюдения на базе БПЛА и стационарных платформ.

В разработанной системе используется модель YOLOv8, обеспечивающая обнаружение объектов (люди, техника и другие цели) в реальном времени. Для выявления аномалий применяется дополнительная метрика, основанная на анализе текстурных характеристик изображения, не требующая сложной настройки [66].

Интеграция алгоритмов распознавания с навигационным модулем и системой управления подвесом (gimbal) позволяет определять координаты обнаруженных объектов и автоматически наводить камеру.

Результаты анализа передаются в систему оповещения, обеспечивая оперативную реакцию на выявленные события.

Коды на python (Приложение Г)

Описание модели и компонентов neuro_scout_drone

Общая цель: реализовать интеллектуальный визуальный модуль на борту дрона на Jetson-платформе (или аналогичной), который, детектирует объекты (люди, машины и т.д.) с помощью YOLOv8, обнаруживает аномалии в рельефе по текстуре изображения, получает координаты с GPS-модуля, поворачивает камеру (gimbal) в сторону объекта и отправляет уведомления в Telegram с координатами и событием. Детальное объяснение кода и логики.

Реализован полноценный многоуровневый модуль, сочетающий глубокое обучение (YOLOv8), визуальную аналитику (статистика текстур), геопространственный анализ (GPS), мехатронику (Gimbal через MAVLink), связь с оператором (Telegram API).

Это готовый прототип системы для автономного мониторинга, агоразведки, охраны периметра и поисково-спасательных операций.

Далее необходимо упаковать это в GitHub-проект, сделать версию под Docker на Jetson, добавить Web-панель управления с картой, интегрировать с реальным дроном (ArduPilot / PX4).

Основные модули (Приложение Г)

1. Подключение к RTSP-камере

2. Детекция объектов и аномалий

3. Получение координат GPS
4. Управление Gimbal
5. Отправка тревог

Оптимизация производительности, для этого необходимо использования TensorRT для ускорения инференса на Jetson. Интеграция с ROS необходим для более гибкой архитектуры.

Хранение данных, где реализуется запись видео и логов для последующего анализа и отладки. Безопасность, необходим для проверки, что все соединения защищены, особенно при передаче данных по сети. Репозиторий проекта Neuro Scout Drone , в него входят модули для детекции объектов с YOLOv8, определения аномалий рельефа, управления Gimbal-камерой через MAVLink, получения координат с GPS-модуля, отправки тревожных сообщений в Telegram, захвата видео по RTSP.

Структура проекта:

- detect_and_analyze.py
- camera_stream.py
- gps_tracker.py
- gimbal_control.py
- alert_sender.py
- main_loop.py
- config.yaml

README.md с инструкцией по установке и запуску.

1.requirements.txt.

2.GitHub-оформление (лицензия, структура и пояснение для Jetson).

На рисунке 12 представлена разработанная программа планирования полета, позволяющая организовать аэрофотосъемку территории с заданными параметрами.

Метод аэрофотосъемки широко применяется для оперативного получения данных о земной поверхности. Высокая точность и информативность снимков, полученных с БПЛА, делают этот метод востребованным в науке и промышленности. Съемка осуществляется с высоты от нескольких десятков до сотен метров с помощью фотокамеры, установленной на летательном аппарате. Для создания картографических материалов важно не только обеспечивать необходимое разрешение изображений, но и соблюдать требуемый уровень перекрытия кадров, что достигается за счет точного управления траекторией полета [67].

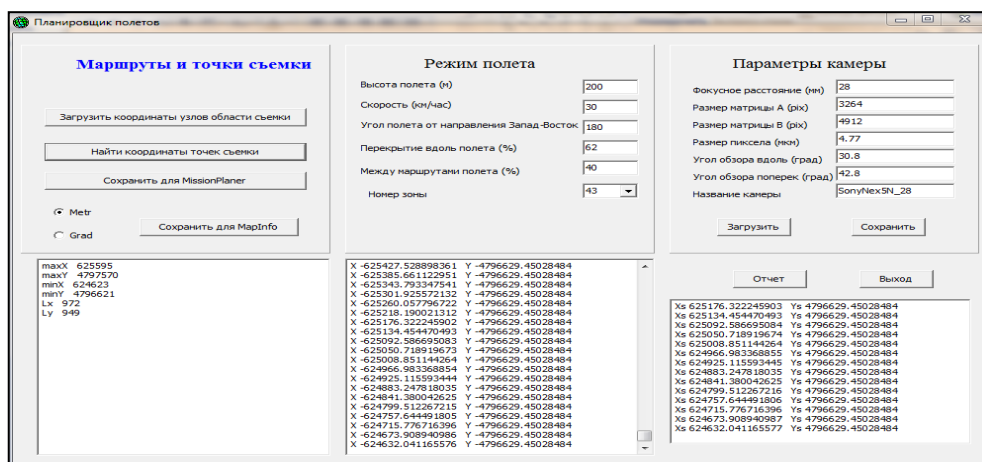


Рисунок 12 – Программа планирования полета

Программа планирования полета использует следующие исходные данные:

- Технические параметры фотокамеры (фокусное расстояние, размер пикселя, разрешение матрицы, углы обзора).

- Координаты границ зоны съемки, задаваемые вершинами полигона.
- Направление полета, выраженное углом относительно оси Восток-Запад.

- Основные параметры полета (высота и скорость).
- Требования к перекрытию кадров (продольное и поперечное).

На рисунке 13 показан пульт управления БПЛА и семиканальный приемник, используемые в процессе выполнения съемки.



Рисунок 13 – Пульт управления БПЛА и приемник на 7 каналов

Во время выполнения аэрофотосъемки на борту БПЛА во флэш-память записывается Log-файл, содержащий сведения о параметрах полета и процессе съемки. Для обработки изображений разработана программа, выполняющая расшифровку и извлечение данных инерциальных измерительных устройств (IMU).

Использование данного решения обеспечивает автоматизацию процесса аэрофотосъемки и повышает точность привязки изображений за счёт учета параметров движения БПЛА.

На рисунке 14 представлена блок-схема алгоритма работы таймера, управляющего срабатыванием затвора фотокамеры, а также его микропроцессорная реализация.

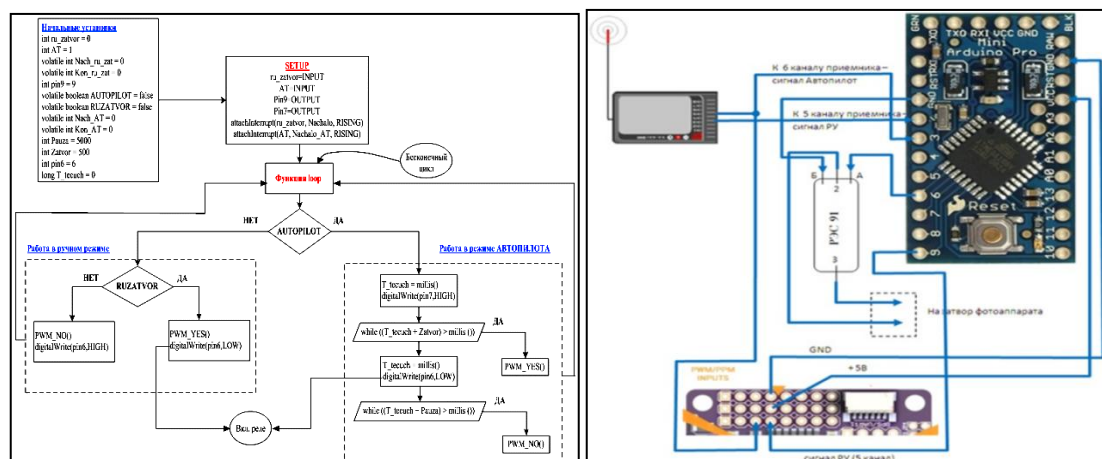


Рисунок 14 – Таймер срабатывания затвора фотоаппарата

- блок-схема алгоритма работы таймера срабатывания затвора фотоаппарата
- микропроцессорная реализация работы таймера срабатывания затвора фотоаппарата

Для планирования аэрофотосъёмки координаты точек съёмки формируются в ГИС (MapInfo или ArcGIS). После проверки и при необходимости корректировки маршрут передаётся в полётный контроллер БПЛА через USB-интерфейс.

Срабатывание затвора фотокамеры управляется программируемым таймером, который работает в двух режимах:

- замыкание контактов затвора внешним управляющим сигналом;
- передача инфракрасного импульса через ИК-передатчик.

При поступлении управляющего сигнала от автопилота на плату Arduino входной сигнал, передаваемый с приемника, блокируется. Затем система генерирует широтно-импульсный сигнал (PWM) с заданным временным интервалом, обеспечивая стабильное срабатывание затвора фотокамеры. Съёмка продолжается до момента отключения управляющего сигнала автопилота.

В стандартном режиме управления с пульта дистанционного управления на приемник поступают PWM-сигналы, которые передаются в бортовую систему АРМ и интерпретируются как команды:

- Автопилот (АТ) – выполнение полета по заданной траектории.
- РУ – ручное управление затвором камеры.
- Дополнительные управляющие сигналы.

После внесения модификаций в систему управления сигналы автопилота и ручного управления получают дополнительную функциональность. При этом

сигнал автопилота одновременно поступает как на АРМ, так и на микроконтроллер, тогда как сигнал РУ передается только на микроконтроллер.

Данное усовершенствование позволило создать модифицированную систему управления БПЛА, обеспечивающую автоматизированное управление аэросъемкой.

Интеллектуальное управление временем срабатывания затвора фотокамеры БПЛА

Во многих задачах, связанных с аэрофотосъемкой с помощью БПЛА, критически важно управлять моментом срабатывания затвора фотокамеры таким образом, чтобы обеспечить равномерное покрытие территории, устранить смазанности изображения (движение БПЛА), съемку конкретных объектов в нужные моменты, синхронизацию с положением БПЛА и внешними условиями (освещенность, облачность и др.).

Интеллектуальное управление предполагает использование алгоритмов машинного обучения, оптимизацию и предиктивный анализ для принятия решений о моменте срабатывания затвора камеры.

Классификация интеллектуальных методов управления

1. Оптимизация на основе маршрута. Использует GPS и инерциальные данные для определения момента, когда камера находится в оптимальной позиции над точкой интереса и осуществляет аэрофотосъемку по сетке.

2. Алгоритмы машинного обучения. Модели (нейросети, SVM и др.) обучаются определять моменты для съемки по входным данным. Мониторинг конкретных объектов.

3. Предиктивные модели (LSTM, Kalman Filter), которые прогнозируют положение БПЛА и ориентацию в будущем, чтобы заранее активировать затвор при высокой скорости полета.

4. Событийно-ориентированное управление. Срабатывание затвора при наступлении определенных условий (обнаружение объекта, высокая NDVI, изменение температуры), для агро- и промышленного мониторинга.

5. Интеграция с цифровыми двойниками. Использует виртуальные модели среды и движения БПЛА для расчета наилучшего времени съемки, применяются при геомониторинге и мультиагентных системах.

Методы и алгоритмы.

Геометрически-информированное управление вычисляет поле обзора камеры (FOV), наклон камеры, высоту полета. Рассчитывается сетка съемки, фиксируется время срабатывания = позиция БПЛА в центре ячейки и поправка на скорость движения.

Алгоритмы: Waypoint-Triggered Shutter, $T_{shutter} = T_{entry} + L_{cell} / 2V$

где L_{cell} - длина ячейки сетки, V - скорость БПЛА.

Нейросетевой предсказатель ориентации использует рекуррентные нейронные сети (LSTM) для предсказания ориентации (pitch, roll, yaw), который

дает прогноз на 1-2 секунды вперед. Затвор срабатывает в момент, когда угол наклона минимален и камера стабилизирована.

Используемые данные: [GPS, скорость, акселерометр, гироскоп]; - [состояние подвеса, угол наклона камеры].

Условно-событийное управление (Fuzzy Logic и Object Detection) применяется если обнаружен интересующий объект на изображении (YOLOv8, SSD, Mask R-CNN), расстояние до объекта уменьшается. Срабатывание камеры при выполнении нечетких условий:

fuzzy

IF object_detected IS True AND distance_to_object IS small THEN trigger_shutter IS now

Многокритериальная оптимизация используется, если нужно учесть несколько факторов: четкость; освещенность; положение по отношению к солнцу; вероятность наличия объектов.

Математическая модель:

$\max [\alpha_1 \cdot \text{Sharpness}(T) + \alpha_2 \cdot \text{Lighting}(T) + \alpha_3 \cdot \text{ObjectProbability}(T)]$

Решается с помощью генетического алгоритма или многокритериального PSO.

Управление затвором по предсказанию позиции (LSTM-подобная логика) Python (Приложение Д).

План и исходный код сцены в Unity, реализующей интеллектуальное управление затвором камеры дрона (симуляция с кнопкой, траекторией и визуализацией срабатывания).

Концепция сцены. Unity-сцена включает дрон (GameObject с траекторией), камеру (на Gimbal или встроенная), кнопку "Снять кадр", интеллектуальное срабатывание (по координатам, времени или детекции объекта), сохранение "снимка" / флаг кадра на сцене.

Эффект: CaptureEffect (вспышка)

Создан Prefab:

ParticleSystem или UI Text "Captured" и анимация исчезновения.

Используется в Instantiate(captureEffectPrefab, ...)

Настройка UI

1. Создан Canvas - Button

2. На кнопку добавлено OnClick() - Drone - выбрано PathController - TriggerShutter()

Проект (Unity .zip), для этого сгенерирована сцена (Unity package);

Добавлена RTSP/Yolo-интеграция;

Интеграция с GPS и координатами (с DEM или предсказанием в LSTM).

Расширенный цифровой двойник съемки

-Имитационная модель полета, движения камеры и освещенности;

-Определяет оптимальные точки и моменты съемки;

-Используется Reinforcement Learning (обучение с подкреплением) для выбора времени:

python

reward = +1 if good_photo else -1

Техническая реализация

-Платформа: Jetson Nano / Raspberry Pi / STM32

-Язык: Python / C++

-Библиотеки: OpenCV, TensorFlow/PyTorch, DroneKit, ROS

-Модули: GPS, IMU, камера с внешним управлением (Sony с API)

Интеллектуальное управление временем срабатывания затвора фотокамеры позволяет существенно повысить качество съемки, уменьшить избыточность данных и автоматизировать полеты дронов при выполнении сложных задач. Методы должны подбираться с учетом контекста: тип полета, цель съемки, технические ограничения и доступные вычислительные ресурсы.

Интеллектуальное управление полетом БПЛА

Разработка аппаратно-программного комплекса для режима автопилотирования БПЛА сопряжена с необходимостью эффективного обмена данными между компонентами системы управления. Организация информационного взаимодействия осуществляется с учетом временных ограничений и приоритетности передаваемых данных [68].

В настоящее время управление полетом осуществляется в двух режимах:

1. Полуавтоматический режим – оператор задает навигационные точки или управляет аппаратом дистанционно.

2. Автоматический режим – управление осуществляется интеллектуальным автопилотом, который выполняет автономное планирование маршрута и корректирует траекторию полета в режиме реального времени.

Тенденция к увеличению уровня автономности БПЛА обусловлена необходимостью снижения зависимости от оператора, а также повышением точности выполнения задач. В связи с этим была разработана система регулирования полета, основанная на применении высокоскоростных микроконтроллеров, обрабатывающих данные в реальном времени.

Основные компоненты системы:

Система БПЛА включает следующие основные компоненты:

- барометрический датчик;
- гироскоп;
- GPS-модуль;
- датчик заряда аккумулятора;
- полётный контроллер.

Ключевые функциональные узлы включают:

- навигационную систему, объединяющую GPS и инерциальные датчики;

- систему регистрации данных полёта.

Автономная система аэрофотосъёмки

Для выполнения съёмки без участия оператора реализован автономный модуль, включающий навигационный приёмник и систему записи данных.

БПЛА может работать в двух режимах:

- дистанционное управление;
- автономный режим.

Автономные беспилотные системы требуют высокой скорости отклика на изменения внешних условий, таких как порывы ветра, разряд аккумулятора, изменения траектории полета. В связи с этим управление полетом возлагается на высокопроизводительные микроконтроллерные устройства, синхронно обрабатывающие сигналы бортовых датчиков и обеспечивающие стабильность полета.

Датчики: считывание и анализ навигационной информации на борту БПЛА
GPS-модуль как средство пространственной навигации

Установка и активация модуля GPS обеспечивает БПЛА базовые функции автономной навигации: удержание позиции, возврат к точке старта и выполнение маршрута по заранее определенным координатам. GPS-модуль становится ключевым элементом управления при полетах в автономном или полуавтономном режиме.

На рисунке 15 представлена антенна GPS, применяемая в конструкции БПЛА.

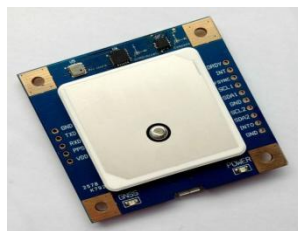


Рисунок 15– GPS антенна, используемая для БПЛА

Барометрический датчик BMP180 используется в составе навигационной системы БПЛА для измерения атмосферного давления и определения высоты полета. Помимо барометрической высоты, датчик способен определять абсолютное, дифференциальное, избыточное давление, а также температуру окружающей среды.

Перед началом эксплуатации датчик должен быть подключен к контроллеру Arduino, откалиброван и снабжен соответствующей библиотекой программного обеспечения. Это необходимо для корректной интерпретации данных, поступающих от сенсора.

Высота полета определяется как расстояние до аппарата, отсчитанное от уровня, принятого за точку отсчета. В зависимости от способа измерения различают следующие виды высот:

Нист. – истинная, от местности под аппаратом,

Нотн – относительная, от условного уровня (взлетной полосы),

Нэш – эшелонная, от изобарического уровня 760 мм рт. ст.,

Набс – абсолютная, от уровня моря.

На рисунке 16 показана классификация высот полета в зависимости от точки отсчета.

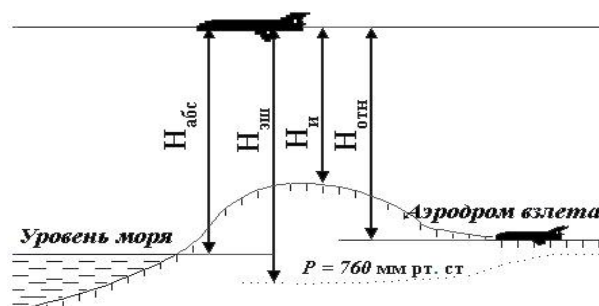


Рисунок 16 – Классификация высот полета по уровню начала отсчета

Для определения высоты полета на основе давления используется барометрическая формула (20):

$$P = P_0 \cdot \exp[(Mgh)/RT] \quad (20)$$

В формуле использовались:

P – давление воздуха на высоте h ;

P_0 – давление на уровне моря;

h – высота над уровнем моря;

$g = 9,8$ – ускорение свободного падения;

T – температура воздуха;

$R = 8.31$ – универсальная газовая постоянная;

M – молярная масса воздуха.

Датчик BMP180 включает в себя мембрану, функционирующую на основе пьезорезистивного эффекта - изменения сопротивления материала под действием деформации. Для подключения используются четыре клеммы: питание (VCC), интерфейсы передачи данных (SDA и SCL) и общий контакт (GND). Возможность работы с напряжением до 5 В обеспечивается встроенным стабилизатором напряжения.

На рисунке 17 схематично изображен внешний вид и распиновка барометрического датчика давления.

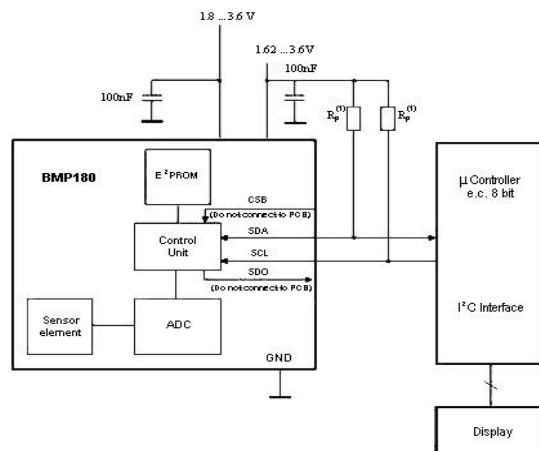


Рисунок 17 – Барометрический датчик давления

Радиотехнический метод измерения высоты основывается на принципе измерения времени между посылкой и приемом электромагнитного сигнала, отраженного от поверхности Земли или воды. Радиовысотомеры обеспечивают определение истинной высоты полета с высокой точностью, особенно на малых высотах. Однако для работы на больших высотах необходимы мощные излучатели и системы защиты от помех.

Определение высоты с использованием GPS осуществляется на основании расстояний до нескольких спутников (от 4 до 6), находящихся на орбитах с известными координатами. Приемник на борту БПЛА рассчитывает географические координаты φ , λ и высоту H относительно модели поверхности Земли.

Для определения трехмерного положения необходимо как минимум три спутника. При увеличении числа спутников точность вычислений возрастает. Преимущество спутникового метода заключается в независимости от атмосферных условий и рельефа местности.

Одним из ключевых элементов навигационной системы БПЛА является гироскопический датчик, отвечающий за определение пространственного положения. Гироскопы применяются для измерения угловых перемещений и вращений объектов независимо от расстояния между датчиком и центром вращения. Особой популярностью пользуются микромеханические гироскопы (MEMS), отличающиеся малыми габаритами, невысокой стоимостью и хорошей интеграцией в бортовые системы БПЛА [69].

Принцип действия вибрационного гироскопа основан на эффекте Кориолиса. При вращении платформы, на которой закреплена вибрирующая масса, возникает сила Кориолиса, вычисляемая по формуле: $F_c = m[\Omega \times v]$, где Ω - вектор угловой скорости, v - линейная скорость, m - масса тела.

Модуль силы выражается как: $F_c = 2m\Omega v \cdot \sin\phi$, где ϕ - угол между векторами v и Ω . Сила направлена перпендикулярно оси вращения и

направлению колебаний, и ее знак зависит от направления движения массы. На этом физическом эффекте основано определение угловой скорости и ускорения [70].

На рисунке 18 представлена конструкция гиродатчика, применяемого для пространственной ориентации БПЛА и приведена конструкция используемого датчика напряжения.

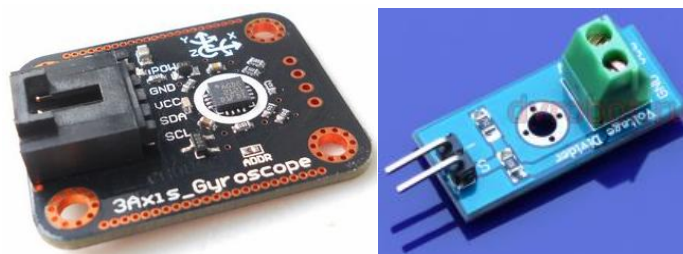


Рисунок 18 – Гиродатчик пространственного положения и датчик напряжения (ID:18)

Контроль за состоянием электропитания реализуется посредством датчика заряда аккумуляторов, функционирующего на базе резистивного делителя напряжения. Такой модуль позволяет в режиме реального времени контролировать напряжение питания до 25 Вольт постоянного тока. Он подключается к микроконтроллеру Arduino и отображает данные на внешнем экране или LCD-дисплее.

Характеристики устройства:

- Рабочее напряжение: 5 В DC;
- Диапазон измерений: 0,0245 В – 25 В;
- Совмещенный аналоговый выход «S» и положительный вход;
- Вес: около 10 г;
- Выводы подключения:
- VCC - положительный контакт измеряемого напряжения (до 25 В),
- GND - общий провод;
- S - аналоговый выход, подключаемый к A0 на Arduino;
- подключение к GND микроконтроллера.

Полетные микроконтроллеры: архитектура и функции

Функционирование систем управления БПЛА обеспечивается полетным контроллером - основной управляющей платой, основанной на микроконтроллерах, таких как Atmega328, Atmega2560 или более производительных STM32. Эти устройства играют роль центрального вычислительного узла и выполняют широкий спектр задач:

- стабилизация полета и положения аппарата в пространстве;
- поддержание заданной высоты (с помощью барометра) и позиции (на основе GPS);

- выполнение маршрута по заранее заданным точкам (при наличии соответствующего ПО);
- передача телеметрических данных на наземную станцию через радиомодуль или Bluetooth;
- обеспечение безопасности за счет возврата к точке взлета при потере сигнала или разрядке аккумулятора;
- поддержка подключения дополнительных устройств: модулей OSD, светодиодов и др.

Количество реализуемых функций зависит от модели полетного контроллера и установленной периферии: в бюджетных решениях часть функционала может отсутствовать.

Система управления БПЛА на основе микроконтроллера ATMEGA

В системе управления БПЛА используется микроконтроллерная платформа ATMEGA, обеспечивающая обработку данных сенсоров и выполнение управляющих воздействий.

Программная реализация выполняется в среде Arduino IDE, что упрощает разработку и отладку алгоритмов управления.

Система функционирует в двух режимах:

- ручное управление с пульта оператора;
- автономный режим, при котором БПЛА выполняет полёт по заданному маршруту с учётом данных сенсоров.

Использование микроконтроллерной архитектуры позволяет адаптировать систему управления под прикладные задачи, включая мониторинг территорий и оперативную оценку состояния объектов.

БПЛА, оснащённые визуальными и инфракрасными сенсорами, могут применяться для наблюдения и поиска объектов в сложных условиях, включая задымление и ограниченную видимость. [71].

Принципиальная блок-схема реализации управления в режиме автопилота представлена на рисунке 19.

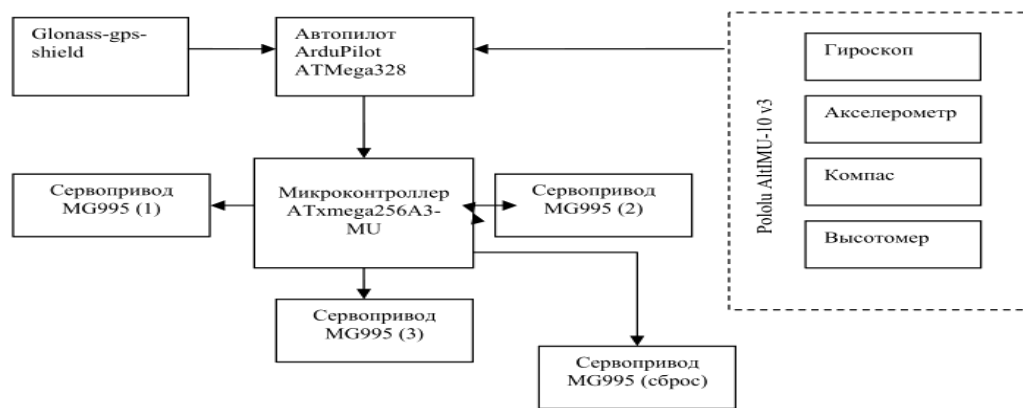


Рисунок 19 – Блок-схема системы управления БПЛА

Прогнозирование поведения беспилотных летательных аппаратов с помощью искусственного интеллекта

Современные беспилотные летательные аппараты (БПЛА) функционируют в сложных, динамически изменяющихся условиях, характеризующихся высокой степенью неопределенности и многомерностью данных. Для обеспечения их надежной и безопасной работы особое значение приобретает задача прогнозирования поведения БПЛА, которая направлена на предсказание будущих состояний аппарата на основе текущей информации о его параметрах, внешней среде и управляющих воздействиях.

Одним из наиболее эффективных направлений решения данной задачи является применение методов искусственного интеллекта (ИИ), способных извлекать сложные зависимости из больших объемов данных и адаптироваться к изменяющимся условиям. Использование ИИ позволяет перейти от традиционных аналитических моделей к обучаемым системам, основанным на машинном и глубоком обучении, что значительно повышает точность и устойчивость прогнозов.

Процесс прогнозирования поведения БПЛА с применением искусственного интеллекта включает несколько последовательных этапов:

1. Сбор и предварительная обработка данных.

На данном этапе осуществляется интеграция данных от бортовых сенсоров (GPS, IMU, барометра, камеры, LiDAR), систем дистанционного зондирования Земли (ДЗЗ) и геоинформационных систем (ГИС). Проводится очистка данных, нормализация и временная синхронизация, формируется обучающий набор признаков.

2. Моделирование и обучение прогнозных моделей.

В качестве базовых инструментов используются модели глубокого обучения - рекуррентные нейронные сети (LSTM, GRU), одномерные и двумерные сверточные сети (CNN), а также архитектуры типа Transformer, обеспечивающие обработку пространственно-временных зависимостей.

Обучение осуществляется на основе исторических данных о траекториях полета, воздействиях и внешних условиях, с минимизацией функции ошибки (RMSE или MAE).

3. Формирование прогноза и оценка неопределенности.

На основе текущего состояния аппарата S_t и управляющих воздействий U_t формируется прогноз будущего состояния:

$$S^{t+\Delta t} = f_{AI}(S_t, U_t, E_t)$$

где E_t описывает внешние факторы (погодные условия, рельеф, плотность препятствий).

Для повышения надежности применяются методы оценки неопределенности (Uncertainty Quantification), позволяющие учитывать степень доверия к полученному прогнозу.

Интеграция с системой управления. Прогнозируемые траектории и вероятности отклонений используются в контуре интеллектуального управления БПЛА для адаптации управляющих воздействий, предотвращения рисков столкновений и оптимизации маршрута.

Использование искусственного интеллекта в задаче прогнозирования поведения БПЛА обеспечивает:

- повышение точности предсказаний траекторий и состояний;
- адаптивность к изменению внешних факторов и параметров системы;
- возможность функционирования в режиме реального времени;
- улучшение безопасности и эффективности выполнения полетных заданий.

Разработанная интеллектуальная модель прогнозирования поведения БПЛА формирует основу для создания интегрированной системы поддержки принятия решений, способной обеспечивать автономность и устойчивость функционирования беспилотных систем в сложных средах.

2.4 Интеллектуальное управление временем срабатывания затвора фотокамеры беспилотного летательного аппарата

При проектировании и разработке БПЛА нередко возникает необходимость создания специализированных электронных узлов, в том числе для управления периферийными устройствами, такими как фотокамера. Для реализации данной функциональности требуется последовательное выполнение нескольких этапов: разработка схемы, подбор элементной базы, трассировка печатной платы и последующая интеграция модуля в общую систему управления полетом.

Процесс разработки схем начинается с предварительного эскиза на бумаге, после чего схема переносится в специализированную графическую среду. Для создания схем и печатных плат использовались два программных продукта:

SPlan - программное обеспечение, предназначенное для построения принципиальных схем. Несмотря на ограниченное количество встроенных элементов, платформа позволяет вручную формировать недостающие компоненты и оформлять схемы в соответствии с нормативами технической документации.

Layout - среда для трассировки двухсторонних печатных плат размером до 300×300 мм. Программа поддерживает как метрическую, так и дюймовую системы координат, обладает широким функционалом для построения контактных площадок, геометрических фигур и вычерчивания общих шин.

Также имеется возможность проведения трассировки, тестирования соединений и нанесения текстовых обозначений.

Для получения функционального прототипа была выполнена самостоятельная сборка печатной платы в лабораторных условиях, включающая этапы: подготовка чертежей, обработка фольгированного диэлектрика, нанесение топологии, травление, лужение и сверление [72]. Пример изготовленной платы приведен на рисунке 20.

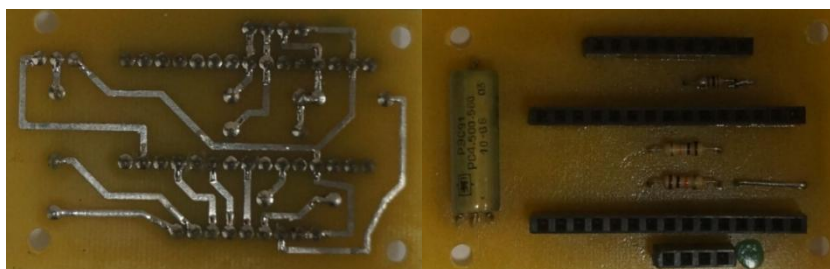


Рисунок 20 – Разработанная плата

Электронные модули и их сопряжение с полетным контроллером. Одним из ключевых компонентов системы является электронный модуль управления скоростью бесколлекторных двигателей, взаимодействующий с центральным контроллером БПЛА. В данной системе применяются регуляторы с функцией Battery Eliminator Circuit (BEC), обеспечивающие стабилизированное питание (обычно 5 В) для приемников, сервоприводов и других компонентов.

В случае отсутствия встроенного BEC может использоваться внешний модуль UBEC, отличающийся возможностью работы с различными видами нагрузки. Преимущество отдается импульсным стабилизаторам (Switching), обладающим высоким КПД и низкой тепловой нагрузкой. При подключении нескольких регуляторов к одному приемнику, целесообразно оставить только один активный стабилизатор, отключив остальные.

Для управления бесколлекторными электродвигателями требуется определение положения ротора. В рассматриваемой архитектуре применяется методика, основанная на анализе обратного электромагнитного поля, генерируемого обмотками при вращении. Это позволяет отказаться от физических датчиков, упрощая конструкцию и снижая вес. Управление скоростью осуществляется посредством широтно-импульсной модуляции (ШИМ), где изменение длительности импульсов определяет величину создаваемого магнитного поля и, как следствие, частоту вращения ротора [73].

Оптимизация компоновки электронных блоков и методы детектирования объектов на изображениях

Компоновка и подключение компонентов управления. Одним из важнейших этапов построения БПЛА является корректное размещение и подключение электронных модулей, в частности регуляторов скорости бесколлекторных двигателей. Подключение осуществляется посредством

трехпроводных соединений. Последовательность подключения этих проводов к мотору не имеет значения, поскольку направление вращения может быть изменено простым перестановкой любых двух проводов.

Для обеспечения надежности соединений рекомендуется использовать разъемы типа Gold Bullet Banana Connector, подбирая их в соответствии с диаметром провода. При необходимости удлинения соединительных кабелей следует соблюдать важные правила:

- предпочтение следует отдавать удлинению линии между регулятором и двигателем;
- использовать провода сечением выше штатного для минимизации потерь.

Удлинение проводов между аккумулятором и регулятором нежелательно, так как индуктивность длинных проводов может вызвать импульсные помехи на входе регулятора. Это может привести к ошибкам в определении положения ротора, сбоям в работе процессора и даже к полному выходу из строя контроллера. Известны случаи повреждения электроники при увеличении длины кабеля от аккумулятора до 30 см. В то же время, удлинение между регулятором и мотором критично лишь для задержки тайминга, которая компенсируется автоматически.

Детектирование объектов на аэрофотоснимках

Детектирование объектов является ключевым этапом анализа изображений в интеллектуальных системах управления БПЛА. Оно позволяет определять наличие, положение и форму объектов на изображении, что важно для построения сцен и принятия решений в реальном времени.

Существуют три основные группы методов детектирования объектов:

1. Методы на основе движения. Эти подходы основаны на сравнении последовательных изображений для выявления движущихся объектов [15]. Однако, для БПЛА, выполняющих съемку неподвижной местности, этот метод не применим, поскольку за время между снимками сцена может существенно измениться.

2. Методы сопоставления с шаблоном. В этом случае имеется эталонное изображение объекта (шаблон), и каждый участок тестового изображения сравнивается с ним. Простейший вариант предполагает сравнение матриц интенсивности цвета, тогда как более сложные используют наборы признаков или вероятностные модели [63].

Сравнение осуществляется «бегущим окном» по всему изображению с расчетом метрик: евклидово расстояние, свертка ошибок, корреляция. Этот метод требует значительных вычислительных ресурсов.

В современных реализациях шаблонный поиск может быть ускорен с помощью GPU, однако потребление энергии не позволяет использовать его на

борту БПЛА. В перспективе планируется применение платформ FPGA, уже реализуемых, в том числе в научных учреждениях (СПбГУ).

3. Методы, основанные на извлечении характерных признаков. Эти методы используют ключевые точки (углы, контуры) или глобальные особенности объектов. Подход включает два этапа:

- Формирование признаков и обучение модели, с использованием алгоритмов машинного обучения;
- Поиск объектов, где признаки тестового изображения сравниваются с обученной моделью, принимается решение о принадлежности объекта заданному классу [74].

Для извлечения признаков может применяться масштабирование и «скользящее окно». Классификаторы (SVM, Random Forest или нейросети) выдают не только метку, но и степень уверенности в принадлежности объекта.

Применение шаблонного поиска и картографирование

Для распознавания объектов на изображениях планируется использовать обученные модели. Хотя этап обучения может быть продолжительным, последующий поиск выполняется с высокой скоростью. С переходом на аппаратные платформы (FPGA) шаблонный поиск станет применимым и для легких БПЛА.

Составление 2D карт

Формирование двухмерных растровых карт возможно путем объединения последовательных снимков, сделанных БПЛА. Эти карты применяются для ориентирования аппарата и навигации. Однако в отличие от вертикальной съемки, широко применяемой в коммерческих сервисах [75], задачи ориентирования БПЛА требуют наклонных ракурсов, что усложняет алгоритмы наложения изображений.

Одна из проблем заключается в необходимости сравнения текущего снимка с эталонной картой местности в реальном времени. Это требует значительных вычислительных мощностей. Необходима оценка относительной высоты полета и наличие объектов, представляющих потенциальную угрозу (препятствий), что требует внедрения алгоритмов пространственной оценки и автоматического управления маршрутом.

2.5 Интеллектуальное прогнозирование рельефа местности как сложного объекта

Рассмотрен подход к позиционированию БПЛА, основанный на минимальном числе опорных точек (не более трех) и реализованный в виде программного пакета. Обработка изображений, полученных с двух объективной камеры, осуществляется с использованием библиотеки OpenCV с задействованием вычислительных мощностей GPU. При обработке 1300 снимков

с большой высоты формируется карта местности размером до 68000×267000 пикселей, занимающая порядка 55 ГБ, а время компоновки карты составляет около 4 часов.

Аэрофотосъемка с применением БПЛА активно развивается. Основными требованиями к камерам являются разрешение не менее 10 мегапикселей, а также наличие предварительной калибровки (фокусное расстояние, параметры дисторсии, геометрические характеристики пикселя) [76].

Оптимизация сопоставления изображений БПЛА и спутников

Позиционирование БПЛА базируется на нахождении проективного преобразования, связывающего изображения, полученные с БПЛА, и спутниковые снимки. Для этого используются алгоритмы выделения и сопоставления особых точек. В рамках реализации использовался алгоритм SURF для детектирования ключевых точек и построения дескрипторов, а также два метода сопоставления - полный перебор и ускоренный поиск ближайших соседей с помощью FLANN [65].

Поскольку спутниковые изображения зачастую имеют очень большие размеры, для оптимизации была использована иерархическая структура данных - пирамида изображений, созданная с помощью утилиты gdal2tiles. Изображение разбивается на фрагменты, каждый уровень пирамиды представляет масштабированную версию предыдущего. Вычисление количества слоев пирамиды производится по формуле (21):

$$N = \left\lceil \log_2 \left(\frac{\max(D, 256)}{256} \right) \right\rceil + 1, \quad (21)$$

где $D = \max(W, H)$, W и H - ширина и высота исходного снимка.

Особые точки и их дескрипторы вычисляются для каждого фрагмента, результаты сохраняются в бинарном виде. Система реализована на основе параллельно-конвейерной архитектуры [77].

Для ограничения вычислительных затрат реализован алгоритм локализации области поиска на основе информации о предыдущем положении БПЛА. При этом сопоставление особых точек происходит только в пределах круга, радиус которого определяется по модели камеры и параметрам предыдущего положения [39].

Оценка положения и ориентации БПЛА

После нахождения пар соответствующих точек вычисляется матрица гомографии, которая затем декомпозируется на матрицу поворота и вектор смещения, определяющие положение и ориентацию БПЛА [68]. Экспериментальные тесты показали, что средняя ошибка определения положения составляла около 0.5 метра, а максимальная - не более 6–7 метров. Погрешность в ориентации не превышала 1° по каждой из осей, при этом

наибольшая стабильность алгоритма наблюдалась при положительном тангаже около 30° [58].

Если доступна карта высот, задача определения положения и ориентации сводится к минимизации функции (24):

$$F(\vec{x}) = \sum_{i=1}^N (\|p_i - M(RP_i + T)\|^2) - \min_{\vec{x}}. \quad (24)$$

где p_i - координаты соответствующих точек на изображении БПЛА, P_i - точки земной поверхности, M - матрица внутренних параметров камеры, R и T - матрица ориентации и вектор положения соответственно.

В случае отсутствия карты высот используется фундаментальная матрица, которая также разлагается на R и T , но определяет положение только с точностью до масштабного множителя k , который уточняется с помощью контрольных наземных точек (GCP) [38].

Использование предыдущего кадра видеоряда

Альтернативный подход использует предыдущий кадр и вычисленные для него параметры. Сначала определяется фундаментальная матрица между двумя последовательными кадрами, затем она декомпозируется, и на основе предыдущих параметров положения T_0 и ориентации R_0 осуществляется предсказание новых (25):

$$P_2 = R(R_0 P + T_0) + kT = (RR_0) P + (RT_0) = R_{res} P + T_{res},$$

где $R_{res} = RR_0$, $T_{res} = RT_0 + kT$. (25)

Повышение устойчивости к сезонным и погодным условиям.

Алгоритмы на базе SURF обладают инвариантностью к изменению яркости изображения, что позволяет сохранять стабильность при смене времени суток или незначительных погодных изменениях. Однако сезонные изменения (весна-зима) могут приводить к изменению текстуры местности, и как следствие - к ухудшению качества сопоставлений. В таких случаях предлагается использовать несколько вариантов спутниковых снимков и выполнять параллельную проверку на соответствие, выбирая наиболее релевантную карту [78].

Разработанные алгоритмы, внедренные в программно-аппаратный комплекс, продемонстрировали свою эффективность в решении задач моделирования полета БПЛА, а также автоматического определения его и положения и ориентации. В рамках дальнейшего развития данных решений необходимо формализовать процедуры генерации траектории полета и внедрить механизм обратной связи, позволяющий корректировать курс в случае отклонений от заданной траектории.

В настоящее время на рынке присутствуют различные полетные контроллеры, способные реализовать функции автопилота. Для проведения аэрофотосъемки с целью картографирования необходимо соблюдение ряда условий:

1. Поддержание постоянного масштаба съемки в пределах допустимого

отклонения 5–10 %, что требует стабильной высоты полета.

2. Обеспечение перекрытия соседних снимков не менее чем на 60 %.
3. Применение выдержки, исключающей смазывание изображения.

Соответствующие ограничения предъявляют требования как к программному обеспечению, так и к аппаратной части БПЛА. Управление осуществляется полетным контроллером с микропроцессорной архитектурой, работающим под управлением встроенной прошивки.

Для аэрофотосъемки требуется возможность полета по заранее заданному маршруту, определяемому системой путевых точек. Одним из популярных решений является контроллер ArduPilot (рисунок 21), разработанный компанией DIY Drones и построенный на базе микропроцессора ATMEGA [44].

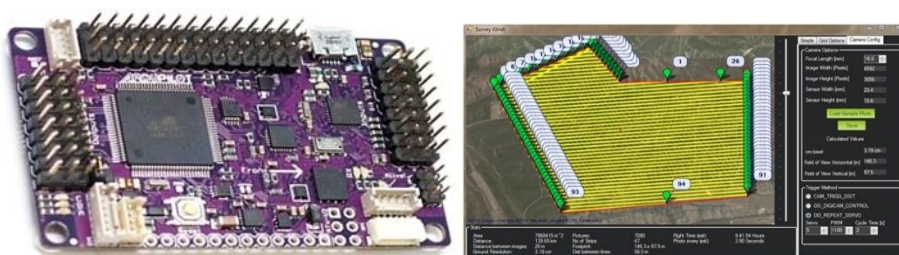


Рисунок 21 – Плата управления ArduPilot

Планирование маршрута и настройка контроллера осуществляется с использованием программы Mission Planner (рисунок 22), позволяющей задать точки срабатывания затвора фотокамеры и загрузить полученные данные в БПЛА.



Рисунок 22 – Расчетные точки срабатывания затвора фотоаппарата

Пример настройки маршрута для заданной территории показывает необходимость выполнения более 7000 снимков вдоль 94 маршрутных линий с шагом 58.5 м, что формирует суммарную протяженность маршрута порядка 139.66 км. Однако реальная продолжительность автономного полета гражданских БПЛА ограничена временем работы аккумуляторов, обычно составляющим 20–30 минут. За этот период БПЛА может преодолеть расстояние не более 2 км, что требует разбивки общего маршрута на несколько автономных миссий.

Для управления фотокамерой SONY NEX-5N используются три элемента управления (рисунок 23): механическая кнопка включения/выключения, кнопка затвора и ИК-сенсор пульта.

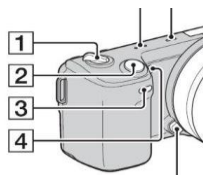


Рисунок 23 – Компоненты камеры

Срабатывание затвора может быть реализовано механическим нажатием сервоприводом, передачей ИК-сигнала или замыканием соответствующих контактов.

Для обоснования параметров аэросъемки приведены расчетные данные в таблице 10:

Таблица 10 – Расчетные значения характеристик аэросъемки для цифрового фотоаппарата SONY NEX 5N

Марка фотоаппарата	SONY NEX 5N						
Фокусное расстояние	F=18 mm,						
Размер пиксела	pix=4.7 мкр.						
Размер матрицы X	3264 pix						
Размер матрицы Y	4912 pix						
Скорость полета	5 м/с						
Выдержка	1/2500 с.						
Величина смаза	0.4 мм						
Расчетные значения параметров снимка							
Высота полета (м)	20	40	60	80	100	150	200
Размер кадра X (м)	17.045333	34.09067	51.136	68.18133	85.22667	127.84	170.4533
Размер кадра Y (м)	25.651555	51.30311	76.9546	102.6062	128.2578	192.386	256.5156
Разрешение на земле (см)	0.5222222	1.044444	1.56666	2.088889	2.611111	3.91666	5.222222
Перекрытие вдоль полета= 60%							
Расстояние между центрами кадров вдоль полета(м)	6.81813333	13.6362	20.4544	27.2725	34.0906	51.136	68.1813

Интервал съемки (с)	1.36362666	2.72725	4.09088	5.45450	6.81813	10.227 2	13.6362
Перекрытие вдоль полета = 80%							
Расстояние между центрами кадров вдоль полета(м)	3.40906666	6.81813	10.2272	13.6362	17.0453	25.568	34.0906
Интервал съемки (с)	0.68181333	1.36362	2.04544	2.72725	3.40906	5.1136	6.81813

Несмотря на широкие функциональные возможности полетного контроллера АРМ, его архитектура накладывает ряд ограничений при выполнении задач картографической аэрофотосъемки. Одним из критических недостатков является ограниченное количество точек маршрута, воспринимаемых контроллером. При попытке загрузить более 255 точек управления, оставшиеся данные записываются в область прошивки, что нарушает ее корректную работу и приводит к необходимости перепрошивки контроллера.

В связи с этим возникла необходимость провести обзор существующих решений по управлению затвором камеры и разработать альтернативные способы реализации данной функции. Анализ тематических дискуссий на профильных форумах [16] показал отсутствие готовых решений. Предлагалось использовать стандартные интерфейсы USB или COM, однако использование полноценного компьютера на борту БПЛА невозможно, и функции управления выполняются непосредственно полетным контроллером.

Ранее процесс аэрофотосъемки с БПЛА рассматривался в ряде научных публикаций [17]. Однако прямые описания механизмов управления затвором отсутствуют. В некоторых работах [12] предполагается, что управление затвором камеры осуществляется через USB-интерфейс при помощи автопилота. Программа Mission Planner, используемая для программирования полетного контроллера АРМ, предоставляет команды для управления съемкой.

В таблице 11 приводится перечень команд, предназначенных для управления затвором фотоаппарата [21].

Таблица 11 – Команды управления затвором фотоаппарата.

Команда	Параметры	Описание
Do-Set-Cam-Trigg-Dist	Dist(m)	Активация затвора через каждые m метров полета
Do-Set-Relay	off (0)/on (1)	Однократно устанавливает один из выходов в 0 (0 вольт) или 1 (+5 вольт)
Do-Repeat-Relay	Repeat # Delay (s)	Переключает выбранный цифровой выход заданное число раз (Repeat#) где Delay (s)

		– кол-во сек. между каждым переключением
Do-Set-Servo	Ser NoPWM	Аналогично Do-Set-Relay, но сигнал – PWM
Do-Repeat-Servo	Ser NoPWM Repeat # Delay (s)	Аналогично Do-Repeat-Relay, но сигнал – PWM Ser No – номер канала PWM – максимальное значение PWM
Do-Digicam-Control	нет	Однократное включение затвора

На практике выполнение команд, таких как Do-Set-Relay, требует дублирования инструкций для подлета к точке и активации камеры, что ускоряет достижение предела в 255 событий. Использование Do-Repeat-Relay также ограничено, так как требует заранее знать количество активаций и влечет за собой непрерывную съемку при длительном сигнале.

Проблемы аналогичного характера были выявлены при использовании команд Do-Set-Servo, Do-Repeat-Servo и Do-Digicam-Control. Для преодоления указанных ограничений была разработана независимая схема активации затвора, базирующаяся на микроконтроллере ATmega328 (Arduino Mini Pro) [22]. После перевода БПЛА в режим автопилота, схема автоматически активирует затвор через заданные интервалы времени. Сигнал может подаваться либо путем замыкания контактов спусковой кнопки, либо посредством инфракрасного импульса на соответствующий сенсор камеры.

Сигнал от микроконтроллера формируется в двух вариантах: как PWM и как одиночный импульс определенной длительности. Для обеспечения корректного взаимодействия с АРМ были соблюдены следующие требования:

- исключение помех в системе сигналов полетного контроллера;
- обеспечение возможности управления затвором замыканием контактов.

На рисунке 24 представлена схема подключения таймера на Arduino Mini Pro к полетному контроллеру АРМ.

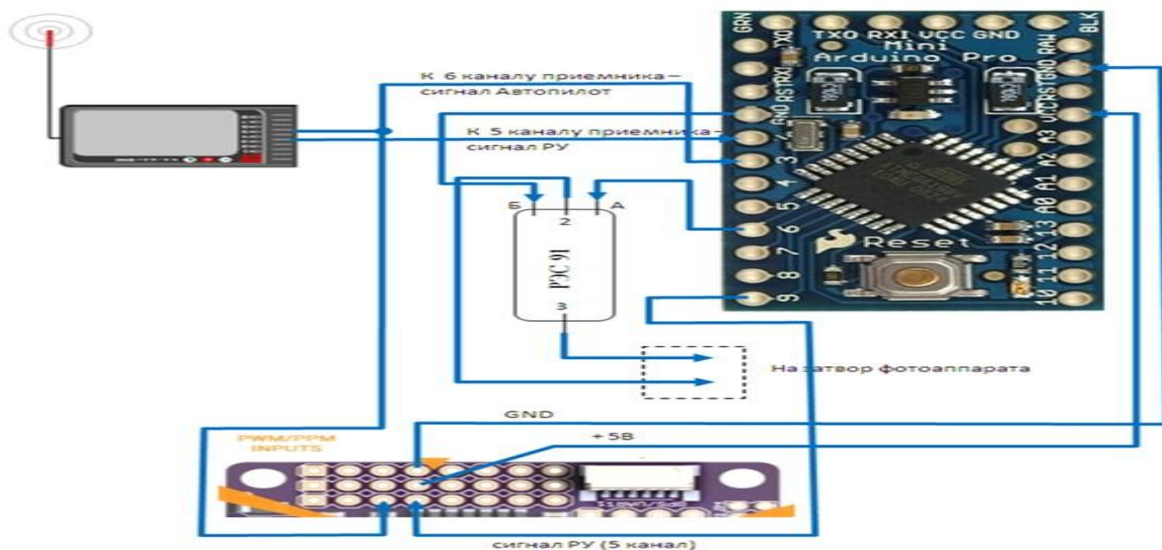


Рисунок 24 – Схема подключения таймера к АРМ

На рисунке 25 показана блок-схема алгоритма работы таймера.

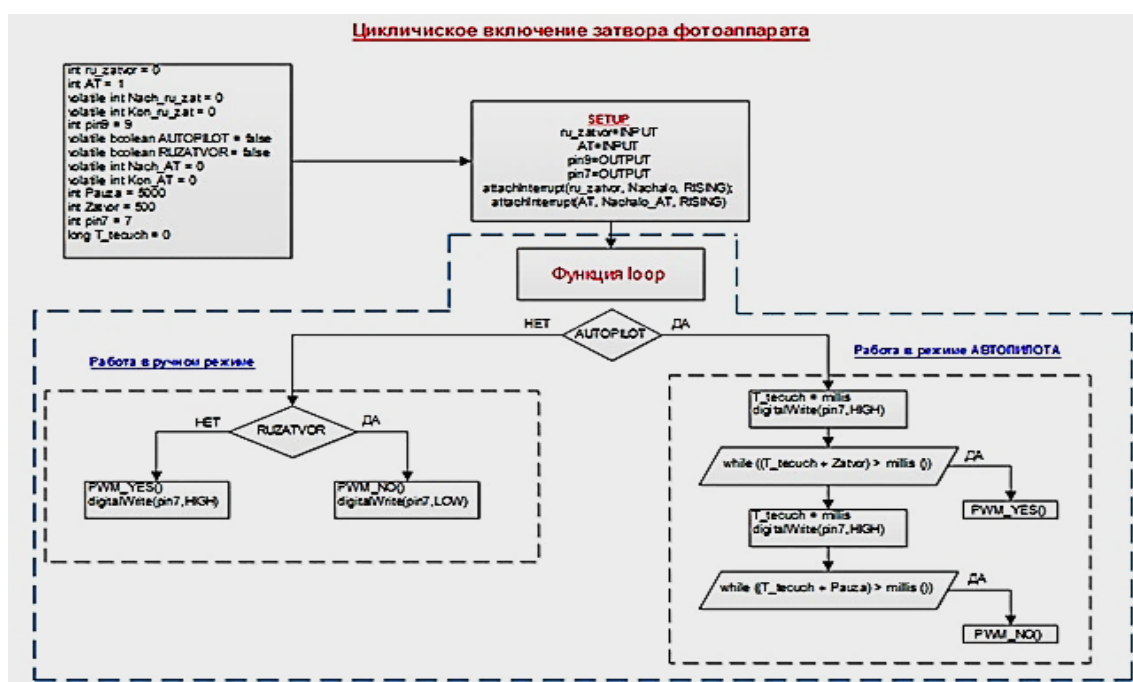


Рисунок 25 – Блок-схема алгоритма работы таймера

На рисунке 26 приведены функции и прерывания, задействованные в основном цикле программы.

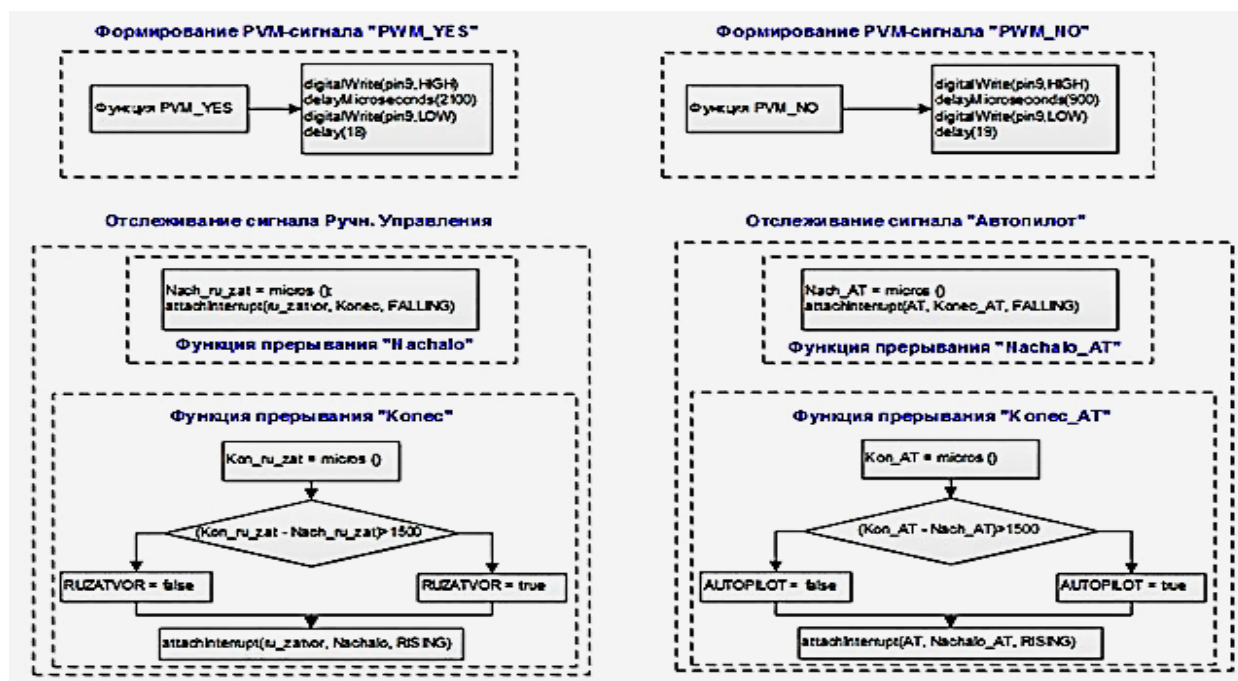


Рисунок 26 – Используемые функции и прерывания.

Реализация данной схемы позволила обойти программные ограничения контроллера АРМ и обеспечила надежную синхронизацию затвора камеры с процессом полета без ущерба для основной логики управления БПЛА.

Описание работы таймера.

В стандартном режиме функционирования система управления БПЛА использует сигналы ШИМ (PWM – Pulse-Width Modulation), поступающие от передатчика на приемник, а затем на плату полетного контроллера АРМ. Эти сигналы управляют различными функциями:

- Автопилот – команда на следование по заданной траектории;
- РУ – команда на ручное срабатывание затвора;
- другие пользовательские команды.

Сигналы «ДА» и «НЕТ» являются цифровыми по сути, однако отличаются по длительности импульса. На рисунке 27 представлена форма сигнала «НЕТ», а на рисунке 28 – сигнала «ДА».

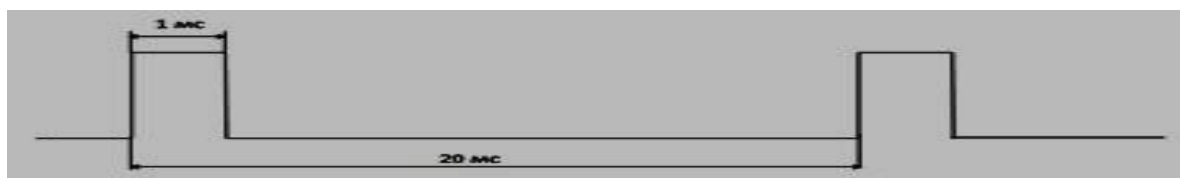


Рисунок 27 – Форма сигнала «НЕТ»



Рисунок 28 – Форма сигнала «ДА»

Сигналы поступают на входы платы Arduino: сигнал Автопилот - на вход 3, а сигнал РУ - на вход 2. При этом сигнал Автопилота подается также на АРМ 2.8, тогда как сигнал РУ обрабатывается только микроконтроллером. На выходе 9 Arduino формируется сигнал PWM, предназначенный для фиксации координат срабатывания затвора в памяти АРМ. Сигналы PWM (выход 9) и аналоговый (выход 7) вырабатываются в зависимости от логики включения:

- При отсутствии управляющих сигналов – формируется сигнал выключения затвора (PWM = 0, реле обесточено);
- При поступлении сигнала РУ – затвор срабатывает вручную;
- При активном сигнале Автопилота – система переходит в автоматический режим, и срабатывание затвора происходит по заданному временному интервалу.

Это позволяет существенно уменьшить количество маршрутов в плане полета, так как сохраняются только ключевые точки, а съемка ведется с заданным шагом, обеспечивая необходимое перекрытие кадров.

Для устранения ограничений штатной прошивки и повышения удобства работы в полевых условиях, был разработан программатор интервала срабатывания затвора, работающий по интерфейсу I2C.

Принципиальная схема блока управления представлена на рисунке 29.

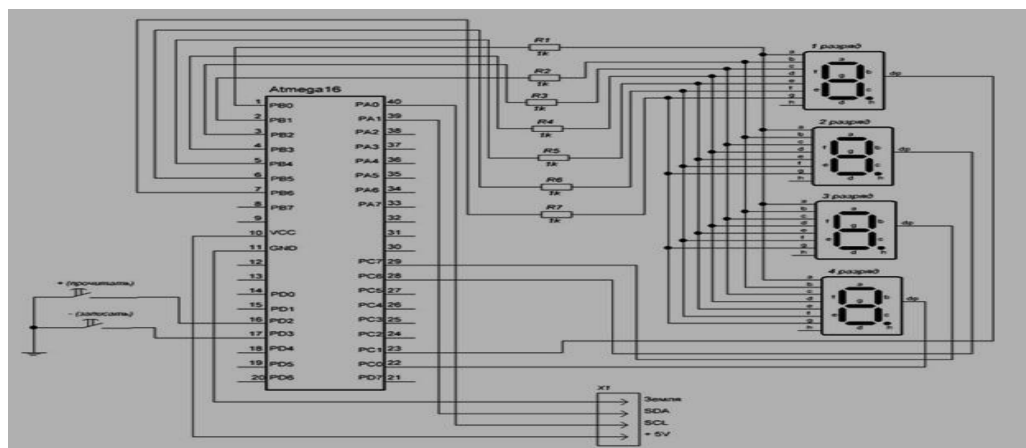


Рисунок 29 – Схема программатора таймера.

Данный блок выполняет следующие функции:

- считывание текущего интервала срабатывания;
- изменение значения интервала (шаг – 50 мс);

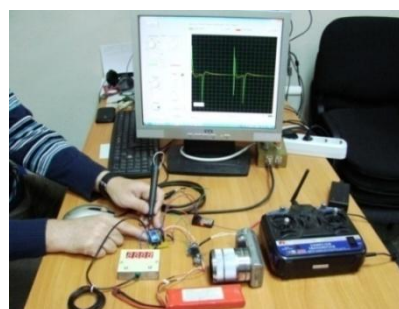
- запись нового значения в энергонезависимую память микроконтроллера.
- Программирование осуществляется с помощью двух кнопок:
- «+» – считывание, удержание >1.5 сек;
 - «-» – запись после установки нового значения.

Связь между программатором и управляющей платой осуществляется четырехжильным кабелем, который передает данные и питание. Работу интерфейса I2C обеспечивают подтягивающие резисторы 10 кОм, размещенные на плате управления.

Проверка корректности работы таймера осуществляется с помощью осциллографа. На рисунке 30 показано рабочее состояние таймера при выставленном интервале 2 секунды и соответствующая осциллограмма сигнала.



а) таймер с программатором и выставленным интервалом 2 секунды



б) осциллограф

Рисунок 30 – Таймер и осциллограф

Затвор камеры может срабатывать двумя способами:

- по замыканию контактов реле (длительность импульса – 0.5 сек);
- по инфракрасному сигналу с ИК-светодиода.

Форма сигнала, поступающего на реле, представлена на рисунке 31.

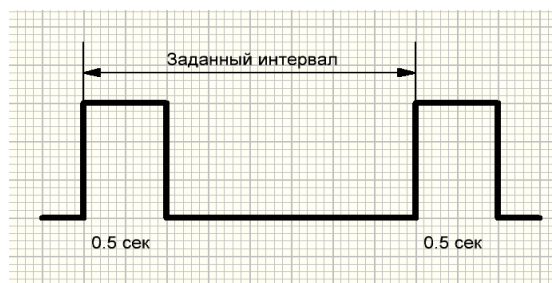


Рисунок 31 – Сигнал, поступающий с микроконтроллера таймера на реле

Осциллограмма, отражающая реальные электрические процессы в цепи реле, приведена на рисунке 30. Ввиду особенностей осциллографа, подавляющего постоянную составляющую, на графике отображены только передний и задний фронты импульсов.

Осциллограмма рабочих сигналов таймера:

- 1 - передний фронт импульса;
- 2 - задний фронт импульса;
- 3 - заданный интервал (пример: 2 секунды)

Испытания работы схемы таймера с подключенным фотоаппаратом

В ходе экспериментальных испытаний были протестированы два режима срабатывания затвора цифровой фотокамеры: посредством замыкания контактов реле и через инфракрасный канал. На рисунке 32 показан внешний вид фотоаппарата, включая элементы, задействованные в управлении затвором.

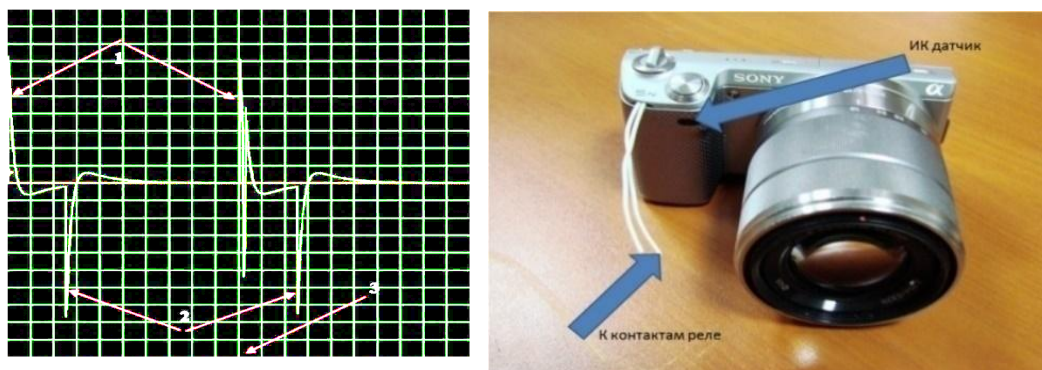


Рисунок 32 – Осциллограмма рабочих сигналов таймера и фотоаппарат с указанием подключения к реле и ИК-датчику

Изменение временного интервала между срабатываниями осуществлялось программно. Установлено, что минимально допустимый интервал, при котором камера гарантированно выполняет снимок, составляет 2 секунды. Этот порог определяется не ограничениями схемы таймера, а особенностями работы затвора самой камеры. Тем не менее, данный интервал полностью обеспечивает необходимое перекрытие снимков (не менее 60%) при стандартных условиях аэросъемки.

В таблице 12 приведены расчетные значения максимально допустимой скорости полета БПЛА при различных высотах съемки с учетом фиксированного интервала между срабатываниями затвора (2 сек) и 60% продольного перекрытия кадров.

Таблица 12 – Расчет максимальной скорости БПЛА

Фокусное расстояние объектива - 18 мм				
Высота (метр)	Размер кадра на земле (метр)	Разрешение	Расстояние между центрами кадров (при перекрытии 60% вдоль полета)	Макс. скорость БПЛА (при мин.

					интервале 2 сек.) м/сек
	X	Y	r (cm/pix)		
50	43.520	65.493	1.333	17.408	8.704
75	65.280	98.240	2.000	26.112	13.056
100	87.040	130.987	2.667	34.816	17.408
125	108.800	163.733	3.333	43.520	21.76
150	130.560	196.480	4.000	52.224	26.112
200	174.080	261.973	5.333	69.632	34.816
225	195.840	294.720	6.000	78.336	39.168
250	217.600	327.467	6.667	87.040	43.52
275	239.360	360.213	7.333	95.744	47.872
300	261.120	392.960	8.000	104.448	52.224

Разработанный программируемый таймер и блок управления затвором камеры подтвердили свою работоспособность при проведении натурных съемок. Созданное решение позволяет использовать цифровые камеры любой модели и обеспечивает необходимую частоту съемки при различных режимах полета БПЛА.

Для решения задачи позиционирования БПЛА по фотоснимкам и их обработки на борту были реализованы следующие ключевые этапы:

- Подготовка цифровой карты местности с выделением характерных объектов;
- Формирование базы данных объектов;
- Построение маршрутного задания;
- Эмуляция выполнения алгоритма обработки и управления.

Для реализации обработки изображений была выбрана библиотека OpenCV с C++ API, которая является промышленным стандартом в области технического зрения. Для хранения информации об объектах применялся формат ShapeFiles, открываемый в популярных векторных редакторах. Реализация основана на библиотеке C_shapefiles.

Проведение полевых испытаний БПЛА

Для проверки функциональности системы аэросъемки, включающей аппаратную часть и алгоритмы управления, была проведена тестовая миссия в пригородной зоне Алматы, в районе поселка Алатау. Участок был выбран по критериям безопасности, так как не имеет плотной застройки (рисунок 33).



Рисунок 33 – Район проведения тестовой аэрофотосъемки: окрестности поселка «Алатау»

Область съемки была ограничена четырьмя опорными точками: A11, A12, A13 и A14. В процессе планирования полета обеспечено продольное перекрытие 60% и поперечное - не менее 30%. Съемка осуществлялась с высоты 200 м. План маршрута съемки представлен на рисунке 36. В результате выполнения миссии было получено 256 снимков с разрешением 16 мегапикселей. Пространственное разрешение на местности составило около 5 см на пиксель, что подтверждается данными на рисунке 34.



Рисунок 34 – План полетных полос и точек съемки и истинный размер объектов на аэрофотоснимке (высота 200 м)

Проведение аэрофотосъемки в районе озера «Сайран» и предварительная фотограмметрическая обработка

В рамках комплексных испытаний созданной аэросъемочной платформы на базе БПЛА был выполнен полет с задачей аэрофотосъемки района вблизи озера Сайран. На рисунке 35 представлена карта района, выбранного для проведения съемки. Поскольку запас емкости аккумуляторной батареи обеспечивал дальность полета не более 1,5 км, полная съемка объекта потребовала выполнения серии полетов с заменой аккумуляторов. Пример запланированных маршрутов движения БПЛА представлен на рисунке 35.

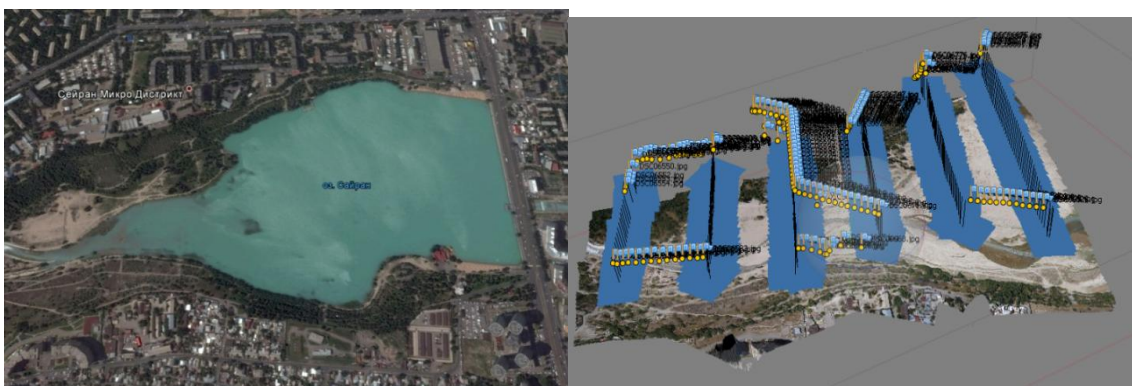


Рисунок 35 – Область аэросъемки в районе озера «Сайран» и схема траекторий запланированного полета

В результате аэрофотосъемки было получено более 300 фотоснимков, обработка и объединение которых выполнены в рамках предварительного фотограмметрического этапа. На рисунке 36 показан результат объединения изображений.

Основы и этапы фотограмметрической обработки. Фотограмметрия позволяет получать метрику и пространственные характеристики объектов на основе анализа их изображений. Съемка может производиться одиночными снимками или сериями перекрывающихся изображений. Преимуществами метода являются высокая точность, безопасность, возможность дистанционного получения данных и автоматизация вычислений.



Рисунок 36 – Мозаичное изображение объединенных аэрофотоснимков

Ключевые этапы обработки включают:

Подготовка исходных данных

На этапе подготовки осуществляется визуальный просмотр полученных кадров, исключение снимков с размытием или выходящих за границы интересующей территории. Пример отобранных снимков показан на рисунке 37.

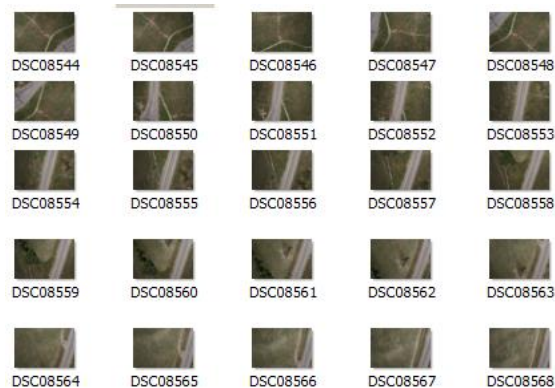


Рисунок 37 – Пример кадров, отобранных для последующей обработки

Также извлекаются координаты центров проекций фотографий, записанные во внутреннюю память БПЛА. Пример фрагмента файла с координатами приведен ниже:

```
DSC08544.jpg 4801662.275 673026.228 884.40
DSC08545.jpg 4801662.539 673026.984 886.41
DSC08546.jpg 4801662.463 673026.613 886.56
DSC08547.jpg 4801660.946 673020.309 885.78
DSC08548.jpg 4801658.007 673009.265 883.59
DSC08549.jpg 4801654.375 672995.019 881.13
DSC08550.jpg 4801651.362 672980.189 879.58
```

Характеристики цифровой камеры

Для точной обработки важно учитывать параметры фотокамеры: фокусное расстояние, координаты оптического центра, физический размер матрицы и пикселя, а также коэффициенты дисторсии. На рисунке 38 иллюстрируется взаимосвязь между фокусным расстоянием, масштабом изображения и высотой съемки.

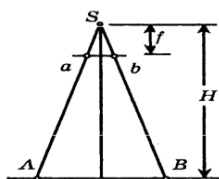


Рисунок 38 – Связь фокусного расстояния размера кадра на земле и высоты съемки (26)

$$\frac{1}{m} = \frac{ab}{AB} = \frac{f}{L} = \frac{f}{H} \quad (26)$$

Здесь AB и ab – расстояния между точками на местности и их изображениями на снимке; f – фокусное расстояние объектива съемочной камеры; H – высота фотографирования; m - масштаб аэросъемки. Процедуры обучения нейронных сетей, используемых для прогнозирования рельефа

Прогнозирование параметров рельефа местности представляет собой важную задачу для навигационных и управляющих систем беспилотных летательных аппаратов (БПЛА). Точность предсказания высотных характеристик поверхности напрямую влияет на безопасность полета, энергопотребление и качество траекторного планирования.

Для прогнозирования рельефа применяется глубокая нейронная сеть, обучаемая на данных дистанционного зондирования и топографических измерений. Объем и структура выборки данных. Обучающая выборка формировалась на основе совмещенных данных:

- цифровой модели рельефа (ЦМР, разрешение 1–5 м/пиксель);
- спутниковых данных Sentinel-2 и LiDAR-съемки;
- телеметрических данных БПЛА, включающих высоту, крен, тангаж и скорость (Таблица 13).

Таблица 13 - Прогнозирования рельефа глубокой нейронной сетью

Параметр	Значение
Количество наблюдений (N)	150 000 фрагментов (по 32×32 точки рельефа)
Доля обучающей выборки	80 %
Доля валидационной выборки	10 %
Доля тестовой выборки	10 %
Диапазон высот	50–1800 м
Дополнительные признаки	Градиент высоты, отражательная способность, тип поверхности

Для уменьшения переобучения применялись методы аугментации данных: поворот фрагментов, отражение и масштабирование, что увеличило эффективный объем обучающих данных примерно на 40 %.

Для прогнозирования параметров рельефа использовались две основные архитектуры (Таблица 14).

Таблица 14 - Архитектура и структура нейронной сети

Архитектура	Особенности	Применение
CNN (Convolutional Neural Network)	6–8 сверточных слоев, ядра 3×3, пулинг 2×2, активация ReLU	Прогноз локальной формы рельефа
CNN-LSTM (гибрид)	CNN для извлечения пространственных признаков и LSTM для учета последовательности координат	Прогноз динамики высот при движении БПЛА

Пример конфигурации гибридной модели:

Вход: блок 32×32 точек рельефа и координаты траектории (x, y, h);

CNN: 4 слоя, 64–128 фильтров, BatchNorm, Dropout (p=0.3);

LSTM: 2 слоя, по 128 нейронов;

Выход: скалярное значение Δh (изменение высоты на следующем шаге).

Функция потерь и алгоритм оптимизации

Для минимизации ошибки прогнозирования использовалась функция потерь MAE (Mean Absolute Error), которая определяется выражением:

$$MAE = 1/N \sum_i$$

Гибридная модель CNN–LSTM показала снижение ошибки MAE на 33–35 % по сравнению с классическими методами.

Модель сохраняет устойчивость при наличии шумов во входных данных.

Модель корректно воспроизводит как абсолютные значения высоты, так и форму рельефа, включая склоны и переходы между участками.

Оптимальной для задачи прогнозирования рельефа является архитектура CNN–LSTM, позволяющая учитывать пространственные и временные зависимости.

Использование MAE в качестве функции потерь повышает устойчивость модели к выбросам.

Для стабильного обучения требуется выборка порядка 100 000 фрагментов.

Дополнительное учёт градиента рельефа в процессе обучения улучшает воспроизведение форм поверхности, что важно при планировании высоты полёта.

Предложенный подход может применяться для прогнозирования других геопространственных параметров.

На этапе подготовки проекта выполняется:

- импорт снимков и параметров камеры;
- загрузка координат центров съёмки и опорных точек;
- настройка системы координат проекта.

Связующие точки представляют собой элементы местности, присутствующие на перекрывающихся снимках. Их определение необходимо для согласования изображений.

Автоматический поиск связующих точек реализован в большинстве фотограмметрических систем. Пример приведён на рисунке 39.

Рисунок 40 – Список и параметры связующих точек после первичной коррекции

Построение ортофотоплана

Комбинируя мозаичное изображение с моделью рельефа, создается ортофотоплан - точное изображение поверхности, в котором каждый пиксель сопоставим с координатами X, Y, Z.

Экспорт данных

Готовые продукты (связующие точки, пикеты, ортоизображения) могут быть экспортированы в сторонние САПР и ГИС-системы для дальнейшего анализа и использования (Рисунок 41).



Рисунок 41 - Построение ортофотоплана

Эмулирование работы алгоритма

Для проверки корректности функционирования разработанного алгоритма была проведена его эмуляция на основе аэрофотоснимков, полученных с БПЛА, выполненных с ортоориентировкой (угол 90°). Снимки имели разрешение 3200×2400 пикселей и сопровождалась файлами метаданных, содержащими координаты центра проекции и углы ориентации камеры относительно поверхности Земли. На рисунке 41 показан процесс построения ортофотоплана. На выбранных изображениях были вручную выделены четыре объекта: дорога, дрова, дерево и перекресток. Последний из них использовался в качестве эталона для проверки точности вычисления координат с применением ранее описанного алгоритма. Результаты были сохранены в формате shapefile, который поддерживается большинством геоинформационных и графических

редакторов. Для работы с пространственными данными использовалась открытая ГИС.

Поведение сложных объектов как объект моделирования

В контексте управления БПЛА под исследованием поведения сложных объектов понимается выбор последовательности действий, направленных на изменение их внутреннего состояния в ответ на внешние условия. Система принятия решений в таком случае включает в себя следующие ключевые этапы:

- распознавание текущей ситуации,
- анализ и формулировка целей,
- разрешение конфликтов между целями,
- планирование действий для достижения поставленных задач.

Одной из важнейших задач в данной области является повышение эффективности оценки и прогнозирования поведения таких объектов. Особое внимание уделяется построению траекторий движения БПЛА при помощи имитационного моделирования, учитывающего динамику внешней среды, ограничения по энергии и особенности рельефа местности.

Принципы системного подхода

Любой сложный объект можно представить как систему, подчиняющуюся определенным законам. В данном контексте речь идет о:

- законах самоорганизации и иерархической структуры,
- технических и экономических закономерностях (эффективность, эквивалентность, правовая адаптивность),
- целеполагании и управляемости в условиях неопределенности.

Анализ и управление поведением таких систем требуют привлечения методов математического и имитационного моделирования, а также оптимизационных подходов, способных учитывать как количественные, так и качественные характеристики объектов.

Оптимизация проектных решений и методология моделирования

В работе [23] представлено подробное описание процесса оптимизации проектных решений для аналитической оценки вариантов развития ситуации и управления маршрутом БПЛА. Подход основан на применении различных технических инструментов, ориентированных на задачи планирования миссий и оценки их эффективности.

Следует отметить, что одной из ключевых проблем в проектировании и моделировании систем автономного управления является неопределенность параметров внешней среды, а также сложность количественной оценки корректности проектных решений. Однако возможность последующей корректировки и адаптации этих решений в процессе выполнения миссии позволяет достигать высокой степени надежности.

Сравнение альтернативных траекторий в различных условиях (рельеф, препятствия, ограничение по энергии) позволяет выбрать наилучшую стратегию поведения БПЛА.

Методологические принципы прогнозирования

Современные методы прогнозирования поведения БПЛА предполагают наличие априорного набора сценариев, а также ограниченного набора контрольных метрик, характеризующих вероятности наступления тех или иных событий. Основой прогнозирования является систематический подход, позволяющий интерпретировать поведение БПЛА в терминах логической модели и предсказывать возможные сценарии развития событий [64].

Декомпозиция поведения сложных объектов

Предлагаемая технология интеллектуального управления и моделирования поведения БПЛА основана на принципе, допускающем компактное формализованное описание любых объектов, чье поведение обусловлено наличием внутренней модели восприятия внешнего мира.

В качестве прототипа для таких поведенческих моделей часто используется человек, обладающий когнитивными способностями. В прикладной задаче такую модель целесообразно декомпозировать на три основных компонента:

- Формирование и поддержание модели окружающей среды – описание контекста;
- Принятие решений – выбор действий на основе анализа текущей ситуации;
- Исполнение решений – реализация выбранной стратегии поведения.

Распознавание ситуации в данном контексте понимается как процесс сопоставления двух состояний окружающей среды: текущего (реального, ретроспективного или прогнозируемого) и эталонного (предустановленного шаблона). Последний задается в виде ограниченного подмножества мира с параметрическим описанием (пространственное положение, объектная структура, физические характеристики окружения).

Архитектура принятия решений

Система управления сложным объектом (в том числе БПЛА) включает в себя три этапа, как показано на рисунке 42:

- Распознавание ситуации и формулировка целей;
- Разрешение целевых противоречий;
- Планирование последовательности действий.

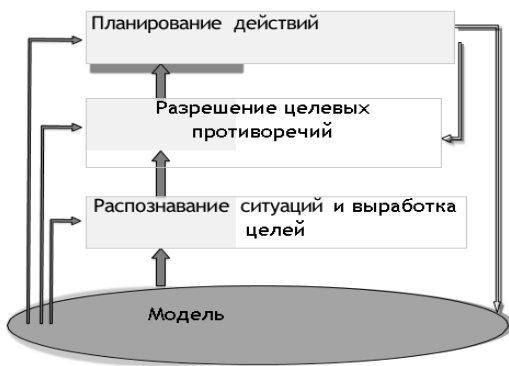


Рисунок 42 – Этапы управления поведением сложного объекта

В условиях многовариантности путей достижения цели необходимо выбрать оптимальный маршрут. Под оптимальностью понимается соответствие заданному критерию (расстояние, время, энергозатраты и т.д.). Планирование действий сводится к поиску оптимального пути на графе состояний.

Математическая формализация оптимального пути

Пусть система имеет множество состояний, представленных вершинами графа. Переходы между ними – дуги, каждая из которых имеет определенную «стоимость» перехода c_{ij} . Тогда критерий оптимальности K_k рассчитывается по формуле (27):

$$K_k = f(c_{ij}); i = 1, \dots, N; j = 1, \dots, N; k = 1, \dots, M. \quad (27)$$

Где: N - количество состояний, M - количество возможных маршрутов, f - функция стоимости маршрута (сумма энергозатрат).

Оптимальный путь определяется как (28):

$$K_z = \text{opt} (K_k) \text{ при } k = 1, \dots, M. \quad (28)$$

Пример планирования траектории для БПЛА

Предположим, БПЛА должен пролететь из текущей точки A в целевую точку B , обходя препятствие - холм с установленной видеокамерой. Интеллектуальная система определяет:

- Время полета по каждому из возможных маршрутов;
- Потенциальную вероятность нехватки аккумуляторной энергии;
- Уровень риска при различных вариантах маршрута.

Стоимость перехода может быть задана как взвешенная функция от времени движения и риска разрядки питания. Система выбирает траекторию, минимизирующую общую стоимость.

Реализация на архитектуре сенсоров

Для оценки состояния среды и определения переменных критериев могут применяться каскады сенсоров, аналогично подсистеме распознавания ситуации. Часто один и тот же каскад сенсоров может обслуживать сразу несколько модулей: принятие решения, планирование и анализ рисков.

Декомпозиция внутри агрегатных систем

В сложных системах, состоящих из нескольких конструктивных единиц (сенсоры, контроллер, механизмы), каждый конструктив ограничен в возможных состояниях. В этом случае планирование поведения может вестись независимо для каждого конструкта, но с учетом ограничений, накладываемых общей системой. Целевое состояние всей системы определяет цели отдельных модулей. На рисунке 43 отражен рельеф местности и границы.

Актуальность для аэрофотосъемки

Применение вышеописанного подхода в аэрофотосъемке обеспечивает:

- Мониторинг состояния рельефа с высокой периодичностью (еженедельно, по участкам в несколько метров);
- Отображение топографии, выделение границ объектов, анализ их изменений;
- Расчет объемов и углов наклона элементов рельефа (насыпи, овраги, здания).

Точное отображение элементов рельефа, объемов и границ критично при строительстве, мониторинге природных рисков, а также в задачах поисково-спасательных операций [71].

Анализ точности и построение ортофотопланов и 3D-моделей рельефа местности

Аэрофотосъемка проводилась с использованием цифровой камеры Sony NEX-5N (фокусное расстояние 19 мм, разрешение 4912×3264 пикселя), установленной на БПЛА. Обработка снимков осуществлялась с применением инструментов Orthophoto Image и Digital Elevation Model (DEM), а финальный монтаж ортофотопланов выполнялся в среде Agisoft Metashape. На рисунке 43 показан результаты работы программы построения накладки монтажа.



Рисунок 43 – Результаты работы программы построения накладки монтажа

Снимки трансформировались и автоматически сшивались в мозаику с перекрытием 80%, при этом фототональное выравнивание производилось автоматически. Для проверки точности ортофотопланов использовались контрольные точки, расположенные в характерных участках рельефа, которые не были задействованы при трансформировании. На каждом фотоплане фиксировалось не менее пяти точек, размещенных на различных высотных уровнях. Максимальное предельное отклонение от расчетного положения составляло не более 0.45 м, что укладывается в допустимые нормы.

Построение 3D-модели и контрольная съемка

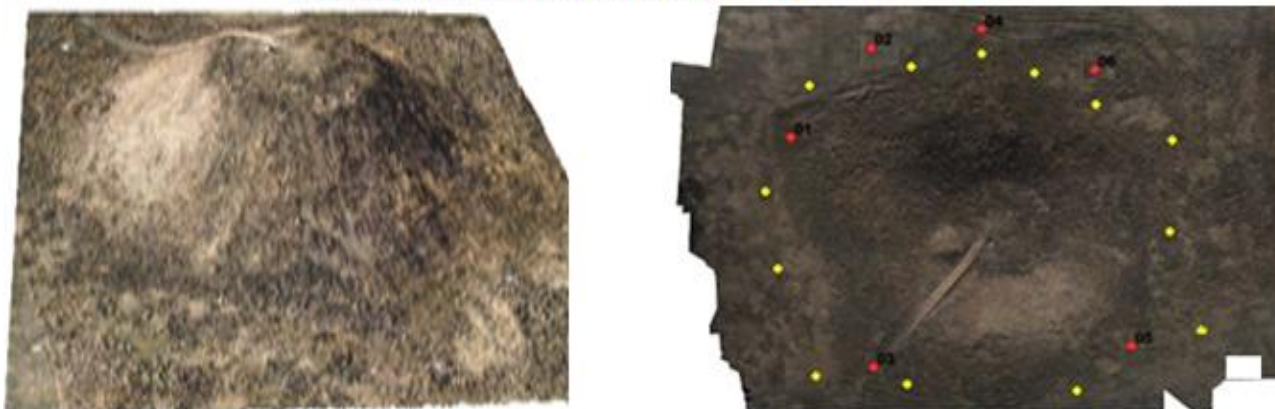
На рисунке 44 представлена 3D модель, реконструированная на основе собранных изображений и опорных точек. Для уточнения траектории облета задавались опорные точки, отражающие позиции срабатывания затвора камеры. На схеме отчетливо визуализированы положения камеры и зона перекрытия снимков.

Обработка включала следующие параметры:

- Количество изображений: 91
- Высота полета: 61 м
- Разрешение: 1.44 см/пикс
- Площадь покрытия: 0.0443 км²
- Количество опорных точек: 91
- Связующих точек: 101,193
- Проекций: 291,265
- Средняя ошибка: 1.56 пикселя

На рисунке 44 отражены 3D модель рельефа местности, Бурулдайского кургана и позиции опорных точек траектории полета БПЛА.

Боралдайские курганы



Обработка включала следующие параметры:

Количество изображений: 91 Высота полета: 61м

Разрешение: 1.44 см/пикс Площадь покрытия: 0.0443 км²

Количество опорных точек: 91 Связующих точек: 101,193

Проекций: 291,265 Средняя ошибка: 1.56 пикселя

Рисунок 44 – 3D модель рельефа местности и позиции опорных точек

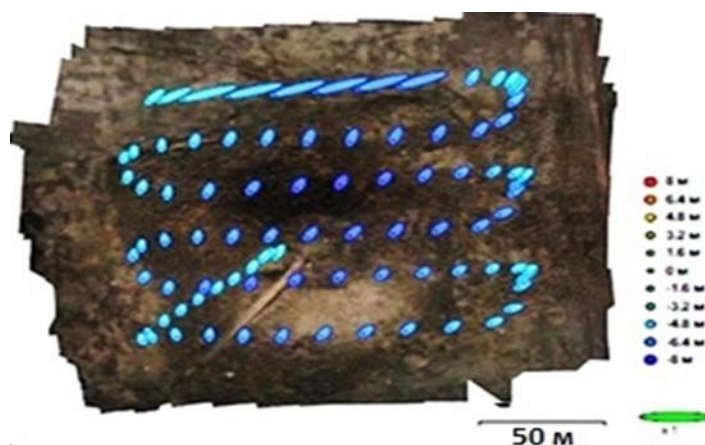


Рисунок 45 - Отражены позиции срабатывания защелки камеры и перекрытие изображения.

Геодезическая точность и анализ отклонений

Для контроля точности применялись опорные и контрольные точки. Их распределение по координатам и соответствующие ошибки приведены в таблицах ниже. На рисунке 46 отмечены позиции опорных (красный цвет) и контрольных точек (желтый цвет).

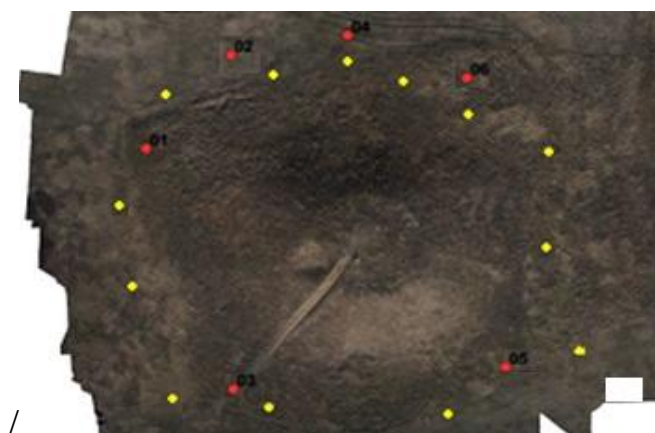


Рисунок 46 – Позиции опорных точек

Эллипсы ошибок визуализируют точность по координатам X, Y и Z (Таблица 15).

Среднеквадратичное отклонение (СКО) используется как метрика качества точности и рассчитывается по классической формуле (29):

$$S = \sqrt{\frac{\sum_{i=1}^n (x - \bar{x})^2}{n}} \quad (29)$$

Таблица 15 – Ошибки отклонения по координатам

X		Y		Z		XY		Общая ошибка
1,6	-1,6	1,6	-1,6	3,2	-3,6	1,6	-1,6	1,6
1,8	-1,8	1,7	-1,62	3,1	-3,8	1,75	-1,71	3,2
2,2	-2,2	1,8	-1,63	3,6	-4,2	2	-1,915	4,65
2,3	-2,3	1,9	-1,64	4,8	-4,4	2,1	-1,97	5,4
4,5	-4,5	2	-1,7	5,2	-3,9	3,25	-3,1	5,7
5,5	-4,6	2,1	-1,72	5,6	-4,4	3,8	-3,16	5,8
6,8	-4,7	2,4	-1,74	5,8	-4,8	4,6	-3,22	5,9
0,9	-4,8	4,2	-1,75	5,9	-5,3	2,55	-3,275	6,7
7,2	-5	4,8	-2	6,7	-6,7	6	-3,5	7,2
7,8	-5,1	4,9	-4,2	6,9	-7,8	6,35	-4,65	7,8
7,6	-5,2	5,2	-4,3	7,2	-8,7	6,4	-4,75	7,86
7,5	-5,3	5,4	-5	8,2	-9,5	6,45	-5,15	7,95
5,6	-4,9	5,1	-4,8	8,0	-9,6	6,5	-5,2	7,96

Опорные точки приведены в таблице 16.

Таблица 16 – Опорные точки

Название	Ошибка, X (см)	Ошибка, Y (см)	Ошибка, Z (см)	Общая (см)	Фото (пикс)
03	-0,98745	-6,08534	2,45898	4,88976	4,26542
04	5,3265	0,690129	-0,1347	2,39361	0,645 (6)
05	0,87456	1,635043	-1,9073	3,69801	0,834 (3)
02	-11,2565	-9,8512	2,13648	3,3167	0,951 (5)
01	5,65897	26,3518	-1,6143	-5,5847	1,537 (9)
06	8,26541	-4,55534	1,94446	10,1711	-2,464
Общая	7,88152	8,185092	2,88364	18,8845	1,80142

В таблице 17 приведено среднеквадратичное отклонение S по опорным точкам, соответственно по координатам X , Y , Z , где общая ошибка составляет 19,12см.

Таблица 17 – Среднеквадратичное отклонение S по опорным точкам

Количество	Ошибка, S_x (см)	Ошибка, S_y (см)	Ошибка, S_z (см)	Ошибка S_{xy} (см)	Общая (см)	Фото (пикс)
	9.18614	16.5288	2.07874	18.91	19.1239	2.667

Возможности анализа на основе ортофотопродуктов

Созданные ортофотопланы и 3D-модели позволяют:

- Измерять площади участков;
- Определять геометрические параметры объектов (длину, ширину, объем);
- Выполнять анализ рельефа, включая уклоны, котлованы, насыпи, курганы и пр.

Полученное облако точек пригодно для построения точных цифровых моделей рельефа (ЦМР), а также для мониторинга изменений рельефа во времени (эрозии почвы, строительных работ, затоплений и т.п.) [79].

Моделирование поведения и оценка траектории

Для оценки эффективности облета по заданной траектории применялся метод имитационного моделирования в контексте игрового противодействия, где взаимодействие элементов рассматривается как поведение в условиях неопределенности. Построенная архитектура моделирования базируется на агрегатном подходе, в котором каждый элемент системы (агрегат) моделируется независимо, но с учетом связей.

Такой подход обеспечивает:

- Универсальность модели;
- Адаптацию к различным сценариям;
- Расширяемость при добавлении новых агрегатов;

– Возможность масштабного описания поведения объектов различного назначения.

Системы, построенные по данному принципу, применимы в организациях, техносфере, робототехнике, городских службах и оборонных структурах [80].

3 Проектирование интеллектуальной информационной системы и разработка прототипа аэросъемочного комплекса

3.1 Концептуальная модель интеллектуальной информационной системы

Современные задачи мониторинга, анализа и прогнозирования поведения сложных динамических объектов, таких как беспилотные летательные аппараты (БПЛА), требуют интеграции разнородных источников данных и применения методов искусственного интеллекта, способных работать с большими объемами пространственно-временной информации. В этом контексте разработка концептуальной модели интеллектуальной информационной системы (ИИС) представляет собой ключевой этап, обеспечивающий методологическое и архитектурное единство всех подсистем.

Интеллектуальная информационная система (ИИС) представляет собой многоуровневую архитектуру, обеспечивающую сбор, хранение, обработку и интеллектуальный анализ данных различной природы - визуальных, сенсорных, геопространственных и контекстных. Основная идея концепции заключается в интеграции технологий глубокого обучения (Deep Learning), физико-информированных моделей (Physics-Informed Neural Networks, PINN) и инструментов геоинформационного анализа (GIS & Remote Sensing Analytics) для повышения точности и достоверности прогнозов поведения сложных объектов в динамической среде.

Система построена по модульному принципу. Взаимодействие между компонентами осуществляется через стандартные интерфейсы, что обеспечивает интеграцию разнородных данных и совместимость с геоинформационными сервисами.

Модуль сбора данных

Обеспечивает получение и объединение данных из различных источников:

- телеметрия БПЛА;
- данные дистанционного зондирования;
- сенсорные и метеорологические измерения.

Данные передаются в унифицированном формате и синхронизируются по времени и координатам.

Модуль хранения данных

Реализует хранение как структурированных, так и неструктурированных данных.

Используется пространственно-временная индексация, обеспечивающая быстрый доступ при анализе.

Модуль обработки данных

Включает методы анализа и прогнозирования поведения объектов.

CNN - используется для обработки изображений и извлечения пространственных признаков;

LSTM - применяется для анализа временных рядов и прогнозирования динамики;

Совместное использование CNN и LSTM позволяет учитывать пространственно-временные зависимости в данных.

Дополнительно могут использоваться более сложные модели (например, трансформеры или физико-ориентированные подходы), однако их применение связано с повышенной вычислительной сложностью.

Для повышения устойчивости прогнозирования используется учет параметров рельефа и условий внешней среды, что позволяет улучшить точность оценки траектории движения БПЛА.

Модуль оценки и прогнозирования - реализует функции адаптивного моделирования и прогнозирования поведения объектов с учетом неопределенностей. Используются методы вероятностного моделирования и оценки доверительных интервалов (Uncertainty Quantification).

При прогнозировании маршрута БПЛА учитывается вероятностное распределение ошибок навигации и внешних возмущений (порывы ветра, температурные градиенты), что позволяет системе адаптивно корректировать курс в реальном времени.

Модуль визуализации и поддержки принятия решений - обеспечивает интерактивное представление пространственных данных и прогнозов в геоинформационной среде (QGIS, ArcGIS, WebGIS). Пользователь может визуально анализировать траектории, тепловые карты риска, зоны турбулентности и прогнозируемые изменения рельефа или растительности [81].

Пример: интеграция с веб-панелью управления (Dashboard) позволяет оператору в реальном времени наблюдать прогнозируемые изменения метеоусловий и оптимизировать маршрут полета БПЛА.

Интеграция с данными ДЗЗ и ГИС

Особенностью концептуальной модели является ее ориентация на использование геопро пространственных данных в контексте интеллектуального анализа. БПЛА выступает в роли локального источника данных высокого разрешения, в то время как спутниковые и ГИС-данные обеспечивают глобальный контекст. Совмещение данных БПЛА и спутников (Sentinel-1/2) позволяет формировать пространственно-временные карты характеристик поверхности. Эти данные используются для выявления изменений, включая деградацию почв и изменения ландшафта.

Применение моделей анализа данных обеспечивает не только обнаружение, но и прогнозирование динамики изменений. Пример интеграции данных приведён на рисунке 47.



Рисунок 47 - Интеграция данных ДЗЗ и ГИС

Интеграция системы с ГИС выполняется следующим образом:

PostGIS / GeoServer - хранение и публикация данных;

QGIS / ArcGIS - визуализация и анализ;

API (WMS/WFS) - передача результатов между модулями системы.

Программная реализация представлена в Приложении Д.

Интеллектуальная интеграция ДЗЗ и ГИС позволяет повысить точность оценки и прогнозирования состояния сложных объектов. Использование гибридных нейросетевых архитектур (CNN и LSTM) обеспечивает синтез пространственно-временных признаков, а ГИС-инструменты позволяют визуализировать и управлять процессами в реальном времени.

Принципы функционирования системы

Концептуальная модель основывается на следующих принципах:

- Интеллектуальная адаптивность - способность системы самообучаться на новых данных и корректировать модели прогнозирования.
- Интеграция разнородных источников данных - совмещение телеметрических, визуальных и геопространственных данных в едином цифровом пространстве.
- Физическая интерпретируемость - обеспечение соответствия результатов моделирования физическим законам.

- Пространственно-временная когерентность - сохранение согласованности данных между временными и географическими масштабами.

- Модульность и масштабируемость - возможность расширения системы за счет подключения новых моделей, сенсоров и источников данных.

Разработанная концептуальная модель может быть применена в задачах:

- управления полетами БПЛА при мониторинге сельскохозяйственных территорий и инженерных сооружений;

- прогнозирования природных рисков (оползни, эрозия, паводки) на основе интеграции данных ДЗЗ и локальных наблюдений;

- оценки состояния инфраструктуры (мосты, дамбы, дороги) по результатам фотограмметрической реконструкции и нейросетевого анализа изображений.

Предложенная концептуальная модель ИИС формирует научно обоснованную основу для последующей разработки архитектуры, алгоритмов и программных решений, обеспечивающих интеллектуальную поддержку анализа и прогнозирования поведения сложных объектов в цифровой геопространственной среде [82].

3.2 Структура и функции основных модулей интеллектуальной информационной системы

Разработанная интеллектуальная информационная система (ИИС), основанная на использовании методов искусственного интеллекта, предназначена для интеграции и анализа данных, поступающих от беспилотных летательных аппаратов (БПЛА), средств дистанционного зондирования Земли (ДЗЗ) и геоинформационных систем (ГИС).

Архитектура системы имеет модульную структуру и включает в себя взаимосвязанные функциональные подсистемы, обеспечивающие полный цикл обработки данных - от сбора до прогнозирования и визуализации результатов.

Модуль 1. Сбор и агрегации данных

Основное назначение данного модуля заключается в обеспечении непрерывного поступления пространственно-временных данных от различных источников:

- БПЛА - получение телеметрических данных, изображений и видеопотоков;

- ДЗЗ - получение спутниковых снимков, данных спектрального анализа и температурных карт;

- ГИС - интеграция пространственных слоев, цифровых моделей рельефа и тематических карт.

Функции модуля: Прием и синхронизация многоканальных потоков данных в реальном времени; Предварительная фильтрация шумов и устранение

пропусков; Агрегация данных по временным и пространственным координатам; Формирование единого потока информации для последующей обработки.

Модуль 2. Хранение и предварительная обработка данных

Модуль предназначен для структурирования, систематизации и подготовки больших объемов данных к анализу.

Он реализует принципы построения гибридного хранилища, включающего базы временных рядов, пространственные базы данных и файловое хранилище изображений. Основные функции: Очистка, нормализация и стандартизация данных из различных источников; Геокодирование и объединение с метаданными; Извлечение и формализация признаков, необходимых для обучения моделей (feature engineering); Формирование обучающих и тестовых выборок; Обеспечение безопасности и резервного копирования данных. Используемые технологии: PostgreSQL/PostGIS, Hadoop, Apache Kafka, Airflow.

Модуль 3. Интеллектуальный модуль анализа и прогнозирования

Этот модуль является ядром системы и обеспечивает реализацию методов искусственного интеллекта для оценки текущего состояния сложного объекта и прогнозирования его поведения.

Подсистемы и функции. Оценка текущего состояния: Распознавание аномалий и оценка технического состояния объекта по данным сенсоров и изображений; Классификация состояний и оценка рисков по отклонениям от нормальных параметров.

Прогнозирование поведения

Использование нейронных сетей (LSTM, CNN, Transformer) для временных и пространственно-временных данных;

Прогнозирование траекторий БПЛА и динамики процессов по данным ДЗЗ и ГИС;

Учет внешних факторов (погодные условия, рельеф, нагрузка и т.д.);

Оценка неопределенности результатов (Uncertainty Quantification).

Обучение и оптимизация моделей:

Автоматический подбор гиперпараметров (AutoML); Инкрементальное обучение на новых данных; Интеграция физических моделей в архитектуру нейросетей (Physics-informed AI).

Модуль 4. Визуализация и управление

Модуль обеспечивает пользовательское взаимодействие с системой и интерпретацию результатов интеллектуального анализа.

Основные функции:

- Визуализация пространственно-временных данных в ГИС-среде (карты, 3D-модели, траектории БПЛА);
- Отображение прогнозов, доверительных интервалов и сценариев;
- Формирование аналитических отчетов и динамических панелей (dashboard);

- Интерактивное управление параметрами моделирования и сценариями прогнозирования;
- Поддержка принятия решений оператором или управляющей подсистемой.

Инструменты реализации: Grafana, QGIS, Plotly Dash, React/Vue.

Модуль 5. Интеграция и обратная связь

Данный модуль обеспечивает взаимодействие между всеми компонентами системы, а также обмен данными с внешними цифровыми платформами и киберфизическими объектами.

Функции:

- Организация API-интерфейсов и протоколов обмена данными (REST, MQTT, OPC UA);
- Реализация контура обратной связи для автоматической коррекции параметров управления БПЛА;
- Поддержка концепции цифрового двойника (Digital Twin) для симуляции и прогнозирования состояний объекта;
- Контроль доступа, журналирование действий и обеспечение информационной безопасности.

Принцип взаимодействия модулей

- Результаты функционирования системы
- Комплексная оценка текущего состояния сложных динамических объектов;
- Прогнозирование поведения и возможных рисков с высокой точностью;
- Сокращение неопределенности при принятии решений;
- Поддержка интеллектуального управления в реальном времени на основе данных БПЛА, ДЗЗ и ГИС.

Численное моделирование - это метод исследования сложных систем и процессов с помощью математических моделей, которые решаются численными (компьютерными) методами.

Оно применяется там, где аналитическое (точное) решение невозможно или слишком трудоемко.

1. Построение математической модели - система уравнений, описывающих поведение объекта (уравнения Навье–Стокса, теплопередачи, движения и т.д.).
2. Дискретизация - переход от непрерывной модели к конечному числу точек (сетке или ячейкам).
3. Решение численными методами - с помощью алгоритмов и вычислительных процедур.
4. Анализ результатов - визуализация, верификация и интерпретация (таблица 18).

Таблица 18 - Основные методы численного моделирования

Метод	Область применения	Пример
Метод конечных разностей (МКР)	Уравнения теплопередачи, диффузии	CFD, термодинамика
Метод конечных элементов (МКЭ)	Прочностной анализ, механика	ANSYS, Abaqus
Метод конечных объемов (МКО)	Гидродинамика, аэродинамика	OpenFOAM
Метод Монте-Карло	Статистика, неопределенность	Симуляция случайных процессов
Агентное моделирование	Социальные, биологические, экономические системы	NetLogo, AnyLogic

Этапы численного моделирования

1. Формулировка задачи. Определение цели, входных параметров, ограничений.
2. Математическая постановка. Уравнения, начальные и граничные условия.
3. Выбор метода дискретизации. Разбиение области на сетку.
4. Численная реализация. Программирование или использование специализированного ПО.
5. Верификация и валидация. Проверка корректности и сопоставление с экспериментом.
6. Интерпретация результатов. Анализ полей, графиков, распределений, визуализация (рисунок 48, 49).

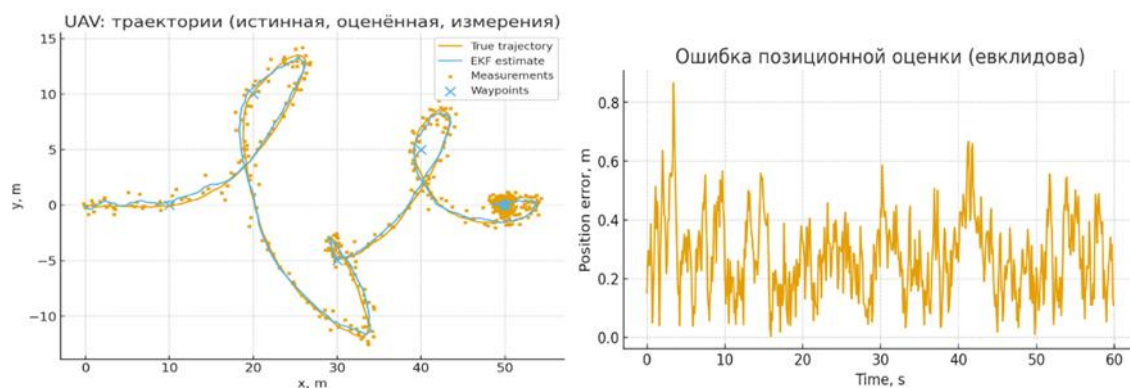


Рисунок 48 – Схема прогнозирующей системы и ошибка позиционной оценки

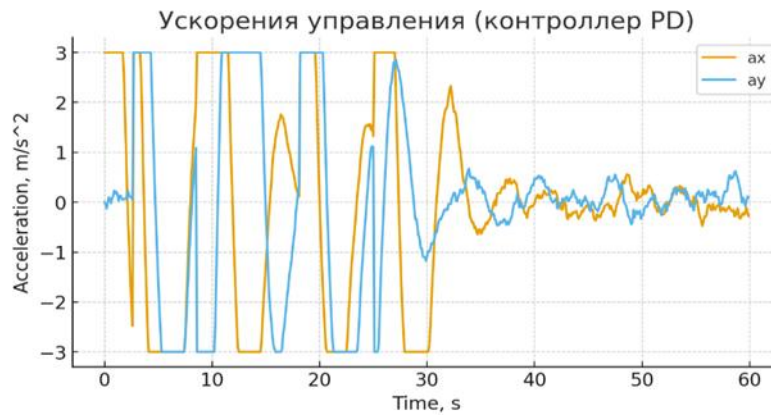


Рисунок 49 – Ускорение управления

Имитация поведение автономного БПЛА (2D), включен простой PD-контроллер для следования за путевыми точками и EKF (Extended Kalman Filter - в данном линейном примере он эквивалентен обычному KF) для оценки состояния по зашумленным измерениям позиции (GPS-подобным). Составлены три графика [63]:

- траектории: истинная, оцененная фильтром и измерения; путевые точки;
- позиционная ошибка оценки во времени;
- управляющие ускорения по осям (a_x , a_y) от контроллера.
- Модель движения - простая модель точки с состоянием $[x, y, v_x, v_y]$ и управлением - ускорения a_x , a_y (матрицы A и B дискретизированы для dt).
- Контроллер - PD (K_p , K_d) вычисляет ускорение по ошибке позиции и скорости; можно заменить на LQR, MPC или более сложный адаптивный контроллер.
- Оценивание - EKF: предсказание по модели и обновление по измерению позиции. Для нелинейных моделей (угловой кинематики, угол рыскания) EKF действительно нужен; для серьезных задач - Unscented KF или PF.
- Шумы - в коде есть шум процесса и измерения; изменяя их, можно изучать устойчивость оценивания и управления.
- Расширения, которые можно добавить: моделирование IMU (акселерометры/гироскопы) и совместная фильтрация (IMU и GPS), моделирование датчика LIDAR/камеры и SLAM, учет ограничений управляющих, планирование пути (A^* , RRT*), уклонение от препятствий, модель аэродинамики и т.д.

Применение численного моделирования

- Инженерия: проектирование конструкций, аэродинамика, гидродинамика.

- Геоинформатика и экология: моделирование эрозии, водных потоков, загрязнений.
- Физика и энергетика: тепловые и ядерные процессы.
- Медицина: моделирование кровотока, биомеханика органов.
- Интеллектуальные системы: симуляция поведения автономных объектов (БПЛА, роботов).

3.3 Алгоритмы функционирования модулей системы

Алгоритмы функционирования модулей информационной системы определяют последовательность обработки данных, обеспечивающую формирование оценок текущего состояния сложного объекта и построение прогнозных моделей его поведения. Каждый модуль реализует набор логических и вычислительных процедур, связанных с потоками данных и контуром обратной связи.

Алгоритм функционирования модуля сбора и агрегации данных

Назначение: получение и унификация данных из различных источников (БПЛА, ДЗЗ, ГИС).

Основные шаги:

1. Инициализация источников данных. Подключение к телеметрическим каналам БПЛА (GPS, IMU, камера, датчики). Загрузка спутниковых снимков и ГИС-слоев.
2. Прием и синхронизация данных. Приведение данных к единому временному формату (UTC). Сопоставление пространственных координат (WGS84).
3. Фильтрация и очистка данных. Удаление шумов и выбросов (методы Hampel Filter, Kalman Filter). Интерполяция пропущенных значений.
4. Агрегация данных. Формирование единого потока пространственно-временных данных; Сохранение в буферную область для последующей обработки.

Результат: интегрированный набор синхронизированных данных для анализа. Алгоритм модуля хранения и предварительной обработки данных.

Назначение: структурирование, нормализация и подготовка данных для обучения интеллектуальных моделей.

Этапы:

1. Импорт данных из буфера сбора;
2. Проверка целостности и метаданных;
3. Очистка и нормализация: Стандартизация единиц измерения (м/с, °С); Удаление дублированных записей.

4. Формирование признаков (feature engineering): Расчет производных показателей (скорость изменения, направление, градиент высоты); Выделение информативных признаков с помощью методов PCA, t-SNE.

5. Разделение на обучающую, валидационную и тестовую выборки.

6. Сохранение подготовленных данных в Data Lake.

Результат: формализованный набор данных, готовый для интеллектуального анализа.

Алгоритм интеллектуального модуля анализа и прогнозирования

Назначение: оценка текущего состояния и прогнозирование поведения сложных объектов с применением методов глубокого обучения.

Основными процедурами являются загрузка обучающего набора данных из хранилища.

Выбор архитектуры модели:

Для временных данных - LSTM, GRU, Transformer;

Для пространственных - CNN, UNet;

Для гибридных - CNN-LSTM или Graph Neural Networks.

Обучение модели осуществляется путем минимизаций функции потерь (MSE, MAE). Оптимизация параметров с использованием Adam, SGD.

Для оценки качества выполняют расчет RMSE, R^2 , и доверительных интервалов прогнозов. Прогнозирование поведения объекта осуществляется путем генерации сценариев будущих состояний. Для оценки вероятностей отклонений и рисков используют адаптивное обучение, что основано на инкрементальном обновлении модели при поступлении новых данных [59].

Результат: прогноз состояния объекта с указанием степени достоверности и неопределенности.

Назначение алгоритма модуля визуализации и управления, это представление результатов анализа в наглядной и интерактивной форме, который состоит из этапов:

1. Получение прогнозных данных от интеллектуального модуля;
2. Визуализация в виде графиков, карт и 3D-траекторий;
3. Отображение пространственных слоев (ГИС);
4. Построение сценариев и сравнительный анализ вариантов;
5. Формирование отчетов и панелей мониторинга;
6. Передача управляющих команд в модуль обратной связи (при автоматическом режиме).

Результат: визуальное представление результатов анализа и прогнозов, поддержка принятия решений оператором.

Назначение алгоритма модуля интеграции и обратной связи, это обеспечение взаимодействия всех компонентов и реализация контура обратной связи между системой и объектом управления.

Основные шаги:

1. Получение выходных параметров от модуля прогнозирования;
 2. Сравнение прогнозных значений с заданными порогами;
 3. Формирование корректирующих воздействий (управляющих команд);
 4. Передача команд на объект (БПЛА) через коммуникационные интерфейсы;
 5. Мониторинг реакции объекта и обновление базы данных;
 6. Адаптация модели на основе новых данных (самообучение).
- В результате получается замкнутый контур управления, обеспечивающий интеллектуальную адаптацию системы к изменяющимся условиям.

3.4 Программная реализация системы

Рассматриваются принципы взаимодействия компонентов и потоков данных.

Эффективное управление комплексом автоматизированных систем в ходе модельного эксперимента на ЭВМ требует постоянного выбора оптимального подмножества элементов и их интеграции в единый технологический процесс. Основная цель - минимизация временных затрат при обработке изделий.

Оценка состояния сложных технических объектов (БПЛА) направлена на определение текущего уровня работоспособности, надежности и эффективности функционирования системы в реальном времени (Рисунок 50).

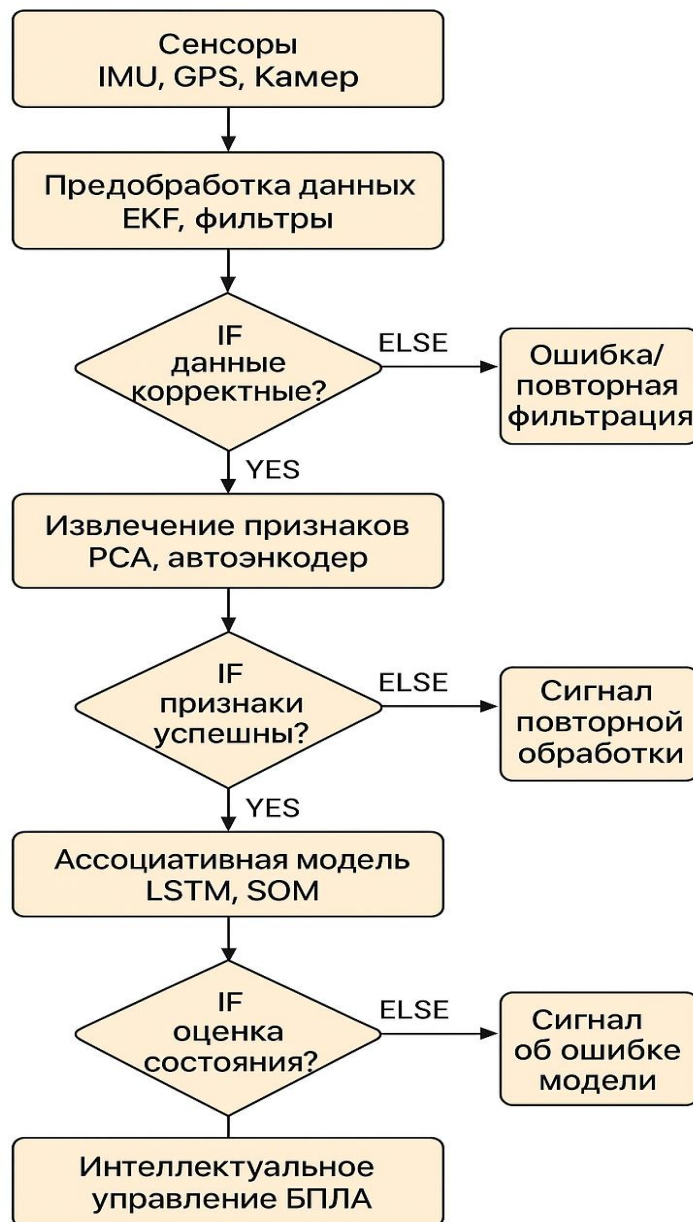


Рисунок 50 - Основные этапы алгоритма оценки состояния сложного объекта

Основные этапы алгоритма оценки состояния

1. Сбор данных. Сенсорные измерения (IMU, GPS, камеры, датчики тока, температуры и т.д.). Параметры среды (ветер, температура, давление)
2. Предварительная обработка. Фильтрация шумов (Kalman / EKF / Particle Filter). Нормализация и устранение выбросов.
3. Формирование признакового пространства. Выделение ключевых признаков (скорость, угол тангажа, уровень заряда, отклонения от маршрута). Использование методов PCA / Autoencoder для уменьшения размерности
4. Модель ассоциативной оценки состояния (АОС). Построение ассоциативной нейронной сети, связывающей текущие параметры с известными

состояниями (норма, сбой, критика). Использование глубокой сети типа LSTM или CNN–LSTM для распознавания динамики состояний

5. Прогнозирование и диагностика. Прогноз будущего состояния на основе временных рядов. Оценка риска отказа и времени до критического состояния

6. Принятие решений (интеллектуальное управление). Автоматическая адаптация управляющих воздействий. Оптимизация маршрута и энергопотребления (Рисунок 51).

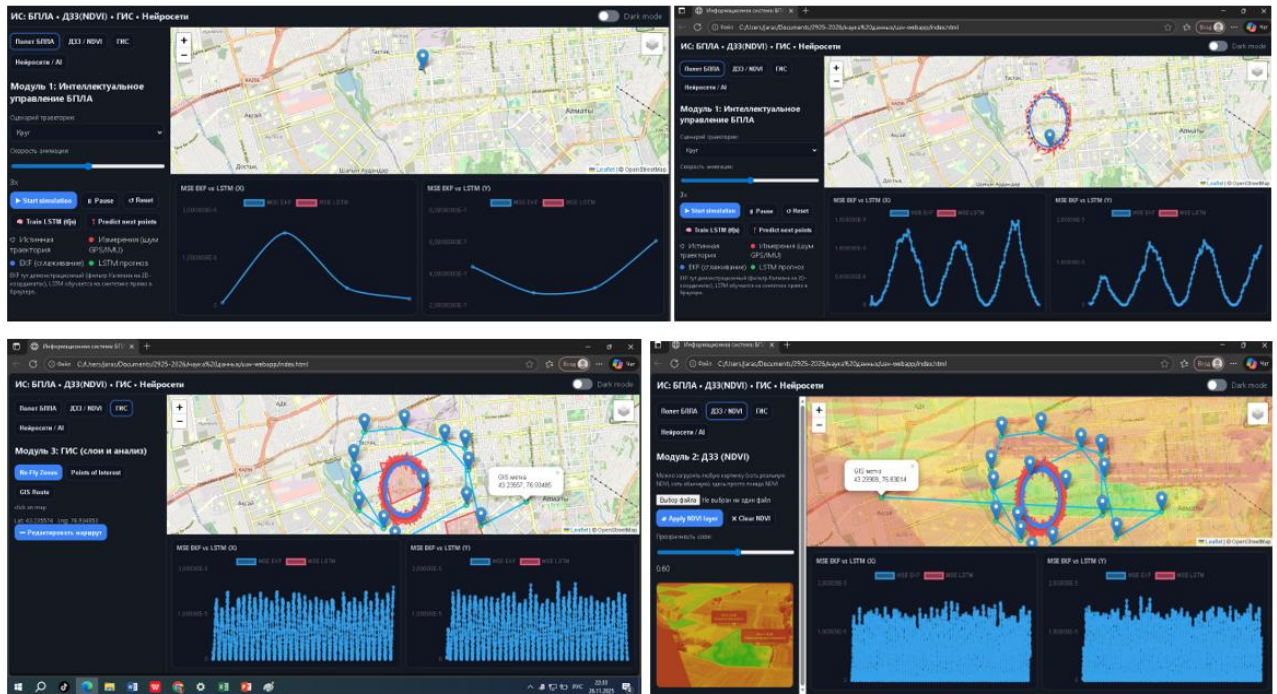


Рисунок 51 - Основные этапы алгоритма оценки состояния

```
import numpy as np
# Генерация гипотетических данных сенсоров (ускорение, гироскоп, GPS)
def generate_sensor_data():
    # Имитация данных с шумом
    true_position = np.array([50, 100]) # Истинное положение
    noise = np.random.normal(0, 1, size=2)
    gps_data = true_position + noise # GPS с шумом
    imu_data = {
        'acceleration': np.array([0.1, 0.2]) + np.random.normal(0, 0.05, 2),
        'gyroscope': np.array([0.01, 0.02]) + np.random.normal(0, 0.005, 2)
    }
    camera_data = None # Для простоты пропустим
    return {'gps': gps_data, 'imu': imu_data, 'camera': camera_data}
```

```

# Простая реализация фильтра Калмана для оценки положения
def kalman_filter(z, x_prev, P_prev, F, H, R, Q):
    # Предсказание
    x_pred = F @ x_prev
    P_pred = F @ P_prev @ F.T + Q
    # Обновление
    y = z - H @ x_pred
    S = H @ P_pred @ H.T + R
    K = P_pred @ H.T @ np.linalg.inv(S)
    x_new = x_pred + K @ y
    P_new = (np.eye(len(K)) - K @ H) @ P_pred
    return x_new, P_new

def main():
    # Инициализация переменных
    x = np.array([0, 0]) # Начальное состояние позицию
    P = np.eye(2) # Начальная матрица ошибок
    F = np.array([[1, 0], [0, 1]]) # Модель движения (статическая для простоты)
    H = np.eye(2) # Матрица наблюдений
    R = np.array([[1, 0], [0, 1]]) * 0.5 # Шум измерений
    Q = np.array([[0.1, 0], [0, 0.1]]) # Шум процесса
    # Цикл повторов до 10 раз
    for step in range(10):
        print(f"\n--- Итерация {step+1} ---")
        # Получение данных с сенсоров
        sensor_data = generate_sensor_data()
        z = sensor_data['gps']
        # Фильтрация данных с помощью EKF/Калмана
        x, P = kalman_filter(z, x, P, F, H, R, Q)
        print(f"Оценка положения: {x}")
        # Оценка состояния (проверка на отклонение)
        deviation = np.linalg.norm(x - np.array([50, 100]))
        if deviation > 10:
            print("Обнаружено отклонение, возможный сбой.")
            # Здесь можно вызвать алгоритмы оценки или сигнал тревоги
        else:
            print("Положение в норме.")
if __name__ == "__main__": main()результат программы:

```

```

--- Итерация 1 ---
Оценка положения: [34.043676 68.18952194]
Обнаружено отклонение, возможный сбой.

--- Итерация 2 ---
Оценка положения: [41.06401684 83.18339158]
Обнаружено отклонение, возможный сбой.

--- Итерация 3 ---
Оценка положения: [44.32727476 89.86208566]
Обнаружено отклонение, возможный сбой.

--- Итерация 4 ---
Оценка положения: [46.57387011 93.74486186]
Положение в норме.

```

В итерации данных представлены оценка поведения, обнаруженное отклонение и возможный сбой (Таблица 19).

Таблица 19 -Критерии оценки эффективности алгоритма

Критерий	Описание	Методы оценки
Точность (Accuracy)	Совпадение оцененного состояния с реальным	Матрица ошибок, RMSE, MAE
Надежность	Устойчивость к шумам и отказам сенсоров	Монте-Карло тестирование, устойчивость фильтра
Энергозатраты	Энергия, необходимая для вычислений и полета	Измерение мощности CPU/GPU и расхода батареи
Вычислительная сложность	Время выполнения и ресурсы памяти	GFLOPS, latency (мс)
Адаптивность	Способность обучаться новым сценариям	Оценка по 5-балльной шкале

Ассоциативная оценка состояния сложных объектов опирается на идею сопоставления текущего состояния с эталонными образами, сохраненными в базе знаний. Формула базовой оценки:

$$S(t)=f(X(t),W)S(t) = f(X(t), W)S(t)=f(X(t),W)$$

где

$S(t)$ - вектор состояния объекта,

$X(t)$ - вектор наблюдаемых признаков,

W - матрица весов ассоциаций, обученная на исторических данных.

Используемые методы: Self-Organizing Maps (SOM) - для визуализации и кластеризации состояний; Deep Autoencoders - для выявления аномалий; LSTM/EKF гибриды - для последовательной оценки и коррекции.

Эффективное управление комплексом автоматизированных систем (на примере БПЛА)

В контексте БПЛА ассоциативная оценка состояния используется для: адаптивной стабилизации полета при сбоях сенсоров; предотвращения перегрева или перерасхода энергии; раннего обнаружения неисправностей двигателей и IMU; оптимизации маршрутов и энергозатрат в реальном времени.

Интеллектуальное управление включает:

1. Автоматическую диагностику состояния узлов.
2. Прогноз отказа с использованием глубинных сетей.
3. Самоадаптацию параметров управления (PID с коррекцией по EKF).
4. Интеграцию с ГИС и ДЗЗ для ситуационной осведомленности.

Интеллектуальное управление БПЛА представлено на схеме, которая иллюстрирует полный цикл работы интеллектуальной платформы для управления БПЛА, обработки данных дистанционного зондирования Земли (ДЗЗ) и использования ГИС-информации. Архитектура логически разделена на несколько слоев: входные данные, обработка, интеллектуальное ядро, интерфейсы и управление дроном.

Оценка технического уровня БПЛА с применением интеллектуальных методов

Оценка технического уровня беспилотных летательных аппаратов (БПЛА) является ключевым этапом выбора эффективных решений при мониторинге, навигации, управлении и интеграции с системами ДЗЗ и ГИС. Для проведения объективного сравнительного анализа выделяются три базовых критерия:

1. Точность - качество выполнения навигационных, измерительных и вычислительных функций.
2. Надежность - способность БПЛА функционировать в условиях неопределенности и внешних возмущений.
3. Энергоэффективность - рациональное использование энергетических ресурсов при выполнении миссии.

Эти параметры позволяют формировать интегральный показатель, характеризующий технический уровень БПЛА по отношению к анализируемым задачам.

Критерии оценки технического уровня БПЛА

Точность (Accuracy Index, AI)

Включает: среднеквадратическую ошибку позиционирования (RMSE); точность стабилизации и удержания высоты; ошибку классификации и сегментации при обработке ДЗЗ.

Формула нормированного показателя точности:

$$AI = 1 - RMSE_{\text{Epos}} / RMSE_{\text{max}}$$

где $RMSE_{\text{Epos}}$ - фактическая ошибка по данным траектории,
 $RMSE_{\text{max}}$ - максимально допустимое отклонение.

Надежность (Reliability Index, RI)

Оценка устойчивости работы БПЛА при внешних возмущениях:

$$RI = T_{stable} \cdot K_{fault} R / T_{mission}$$

где

T_{stable} - время устойчивой работы,

$T_{mission}$ - общее время выполнения миссии,

K_{fault} - коэффициент безотказности (0...1), определяемый по статистике отказов.

Энергоэффективность (Energy Efficiency Index, EEI)

$$EEI = D / E$$

где

D - пройденная дистанция,

E - затраты энергии (Вт·ч).

Для нормирования:

$$EEI_{norm} = EEI / EEI_{max}$$

Интегральный показатель технического уровня. Интегральная оценка:

$$K_{tech} = w_1 AI + w_2 RI + w_3 EEI_{norm}$$

где w_1, w_2, w_3 - весовые коэффициенты (выбираются экспертно или методом АНР).

Пример расчета

Пусть даны измерения для БПЛА А:

- $RMSE_{pos} = 0.42$ м, $RMSE_{max} = 1.2$ м
- $T_{stable} = 18$ мин, $T_{mission} = 20$ мин
- $K_{fault} = 0.96$
- пройдено $D = 6.2$ км
- затрачено $E = 78$ Вт·ч
- $EEI_{max} = 0.12$ км/Вт·ч
- $w_1 = 0.4$, $w_2 = 0.35$, $w_3 = 0.25$

Шаг 1. Точность

$$AI = 1 - 0.42 / 1.2 = 0.65$$

Шаг 2. Надежность

$$RI = 18 \cdot 0.96 / 20 = 0.864$$

Шаг 3. Энергоэффективность

$$EEI = 6.2 / 78 = 0.079$$

$$EEI_{norm} = 0.079 / 0.12 = 0.658$$

Шаг 4. Интегральный показатель

$$K_{tech} = 0.4 \cdot 0.65 + 0.35 \cdot 0.864 + 0.25 \cdot 0.658 = 0.735$$

Итог: БПЛА имеет высокий технический уровень (0.735 из 1).

Адаптивные интеллектуальные методы и используемые метрики обучения

При применении адаптивных нейросетевых моделей для анализа поведения БПЛА и интеграции данных ДЗЗ/ГИС используются следующие методы и параметры, где используемые архитектуры – это LSTM (длинная краткосрочная память) - для временных рядов телеметрии, BiLSTM - для повышения устойчивости прогнозирования, 1D/2D CNN - для локальных пространственных признаков (магнитометр, IMU, карты NDVI), Transformer / Time-Series Transformer - для сложных пространственно-временных зависимостей, CNN–LSTM и CNN–Transformer - гибриды для совместной обработки ДЗЗ и траекторных данных.

Метрики обучения

Для регрессии (траектория, энергия, скорость): MAE (Mean Absolute Error), RMSE (Root Mean Squared Error), MAPE (Mean Absolute Percentage Error), R^2 (коэффициент детерминации)

Для классификации и обнаружения аномалий: Precision, Recall, F1-score, AUC-ROC

Для генеративных моделей (VAE, Autoencoder): Reconstruction Loss (MSE/MAE), KL-divergence, Anomaly Score.

Адаптивные механизмы

Online-learning осуществляет обновление весов в реальном времени. Transfer learning выполняет перенос обучения между различными моделями БПЛА. Domain adaptation - это адаптация к изменению погодных условий / рельефа. Методы устойчивости (robust training) - DropBlock, augmentation, шумы и Uncertainty quantification (MC Dropout, Ensemble) - оценка доверия прогнозу.

Современные технологии моделирования рельефа и анализ полученных моделей играют важную роль в исследованиях, связанных с науками о Земле, экологией, земельным кадастром и инженерными проектами. Использование компьютерных методов обработки пространственных данных значительно расширяет возможности анализа территории, в том числе при изучении загрязнений, моделировании месторождений и планировании устойчивого развития.

Внедрение цифровых технологий в картографию и ГИС произвело революцию в области топографического анализа и визуализации. Современные ГИС позволяют интегрировать цифровые модели рельефа (ЦМР) с тематическими данными, расширяя спектр их применения [45].

Основные методы построения цифровых моделей рельефа

Существует два ключевых подхода к созданию цифровых моделей рельефа:

1. Методы дистанционного зондирования и фотограмметрия. Эти методы обеспечивают высокую точность, но требуют значительных ресурсов и специализированного оборудования.

2. Интерполяция оцифрованных изолиний с топографических карт. Данный метод остается востребованным благодаря доступности исходных данных, хотя он имеет ограничения по точности [36].

Применяемые методы моделирования можно классифицировать по принципу построения моделей. Одним из распространенных способов является построение TIN-моделей, основанных на триангуляции Делоне. Они активно используются в исследовательских проектах, таких как GeoFrance3D [39], а также в задачах интерполяции при нехватке данных [28].

Преимущество триангуляционных моделей заключается в отсутствии искажений исходных данных. Хотя они менее пригодны для детального анализа, их точность остается высокой, поскольку отсутствуют ошибки, характерные для других методов интерполяции. Важные исследования, посвященные этой теме, приведены в работах [49].

Методика моделирования и анализ полученных данных

Разработка методики пространственного моделирования рельефа проводилась с применением стандартных средств ГИС. Основные задачи включали:

- выбор оптимального метода интерполяции;
- дополнение данных в областях с низкой плотностью исходных точек;
- оценку точности построенной модели;
- морфотектонический анализ рельефа.

На рисунке 52 представлена методика, используемая для повышения точности модели в районах с недостатком данных.

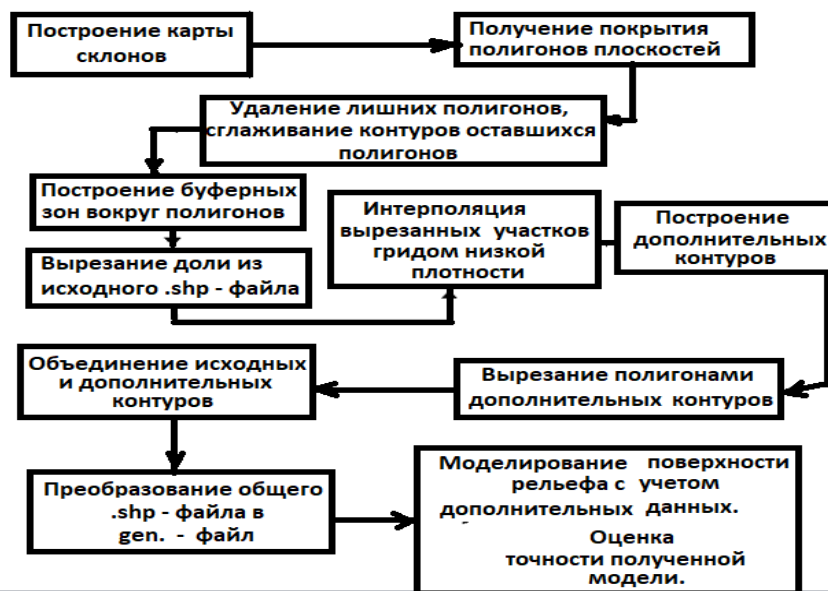


Рисунок 52 – Методика по насыщению данными областей с недостатком опорных точек

Коэффициент корреляции между среднеквадратичной ошибкой исходных и интерполированных данных составил 0.982514, что свидетельствует о высокой точности модели.

Морфотектонический анализ проводился на основе выделения поверхностей выравнивания, которые со временем были изменены под влиянием тектонических процессов. Эти поверхности определялись на основе углов наклона, анализировались тематические карты и строились модели разломов.

На рисунке 53 представлен фрагмент тематической карты, на которой показаны реликты поверхности выравнивания и разломы.

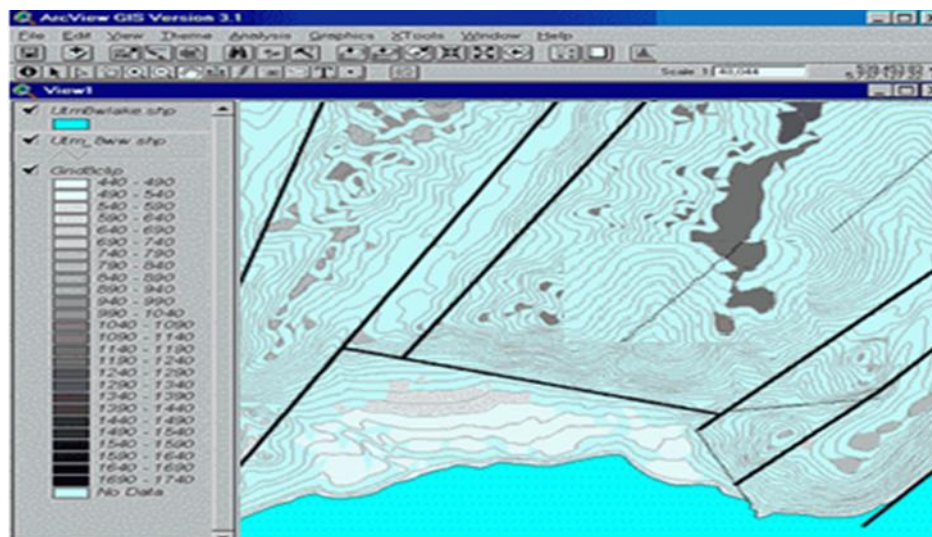


Рисунок 53 – Фрагмент тематической (морфотектонической) карты

В результате моделирования удалось получить поверхность рельефа, соответствующую исходным данным. Разработанная методика позволяет создавать модели, пригодные для дальнейшего географического анализа и использования в ГИС-системах. Морфотектонический анализ показал высокую степень соответствия результатов традиционным методам, обеспечивая при этом более точные количественные характеристики тектонических изменений.

Применение предложенной технологии возможно в различных системах автоматизации, что делает ее универсальным инструментом для научных и прикладных задач.

Алгоритм оценки состояния сложных объектов предназначен для автоматизированного анализа текущего функционирования динамических систем на основе интеграции пространственно-временных, телеметрических и визуальных данных. Его реализация позволяет выявлять отклонения от нормативных параметров, диагностировать техническое состояние и определять риск возникновения нештатных ситуаций.

Повышение точности и оперативности оценки текущего состояния сложных динамических объектов за счет применения методов искусственного

интеллекта, анализа данных БПЛА, дистанционного зондирования Земли (ДЗЗ) и геоинформационных систем (ГИС).

Входные данные.

Телеметрические данные и сенсорные измерения с БПЛА (скорость, высота, углы ориентации, температура, вибрация и др.);

Изображения и видеопотоки с камер БПЛА;

Спутниковые и аэрофотоснимки (ДЗЗ);

Геопространственные данные (рельеф, инфраструктура, погодные параметры);

Исторические данные о работе объекта.

Основные этапы алгоритма

Этап 1. Сбор и синхронизация данных

Подключение к источникам данных (API, сенсоры, базы);

Приведение данных к единому временному и пространственному масштабу (WGS84, UTC);

Формирование интегрированного пакета данных по объекту.

Этап 2. Предварительная обработка

- Очистка данных от шумов и выбросов (фильтр Калмана, медианная фильтрация);
- Интерполяция пропущенных значений;
- Нормализация и стандартизация признаков;
- Объединение гетерогенных данных в единую структуру признаков.

Этап 3. Извлечение признаков и формирование признакового пространства

- Вычисление производных характеристик: скорости изменения параметров, отклонений, трендов;
- Распознавание визуальных признаков на изображениях (сегментация, SIFT, CNN);
- Выделение пространственно-временных закономерностей (градиенты, кластеры, корреляции).

Этап 4. Оценка состояния с использованием моделей искусственного интеллекта

Применение обученных моделей нейронных сетей для классификации состояния объекта:

- Нормальное функционирование;
- Предаварийное состояние;
- Аварийное состояние.

Используемые архитектуры: CNN (для изображений), LSTM/GRU (для временных рядов), гибридные модели CNN-LSTM (для пространственно-временных данных).

Расчет вероятности принадлежности к каждому классу.

Этап 5. Диагностика и интерпретация результатов

- Формирование карты рисков и индекса технического состояния;
- Визуализация данных в ГИС-среде (зоны риска, траектории, аномалии);
- Выдача рекомендаций оператору или управляющей подсистеме.

Этап 6. Адаптация и самообучение

- Сравнение прогнозных и фактических данных;
- Интеллектуальное обновление параметров модели (online learning);
- Оптимизация алгоритма на основе поступающих данных.

Формализованное описание алгоритма

Пусть

$X(t) = \{x_1(t), x_2(t), \dots, x_n(t)\}$ - вектор наблюдаемых параметров объекта,

$Y(t)$ - класс состояния, определяемый моделью.

Тогда процесс оценки состояния описывается как:

$$Y(t) = f(W, X(t))$$

где $f(W, X(t))$ - обученная нейросетевая функция,

W - матрица весовых коэффициентов, определенных в процессе обучения.

Метрика достоверности оценки:

$$P(Y_i|X) = e^{z_i} \sum_j j, \quad i=1, 2, \dots, k$$

где $P(Y_i|X)$ - вероятность принадлежности объекта к классу состояния i .

Классификация объекта в реальном времени дает возможность выявления аномальных режимов и трендов деградации, формирования показателей технического состояния и индексов риска и передачи результатов в модуль прогнозирования и управления.

4 ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ И ОЦЕНКА ЭФФЕКТИВНОСТИ ИНТЕЛЛЕКТУАЛЬНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Быстрое развитие беспилотных авиационных систем (БПЛА), методов дистанционного зондирования Земли (ДЗЗ) и современных геоинформационных технологий (ГИС) создало предпосылки для формирования новых интеллектуальных информационных систем мониторинга территорий и объектов. Традиционные подходы, основанные на ручной обработке данных, характеризуются высокими временными затратами, низкой оперативностью и ограниченными возможностями масштабирования. В этих условиях актуальной научной задачей является оценка эффективности интеграции интеллектуальных алгоритмов обработки данных и пространственного анализа в единую систему управления мониторинговыми процессами.

Целью исследования является определение эффективности реализуемой интеллектуальной информационной системы, основанной на комплексном использовании данных БПЛА, спутниковых снимков ДЗЗ и геоинформационных моделей. Достижение этой цели требует анализа точности, производительности, надежности и экономической результативности системы, а также сопоставления ее характеристик с традиционными методами мониторинга.

4.1 Интеллектуальная оценка и прогнозирование открытых горнорудных месторождений

В качестве экспериментальной платформы для апробации интеллектуального анализа рельефа были выбраны блоки аэросъемки, полученные с применением БПЛА. Все примеры обработки реализованы с помощью профессионального фотограмметрического ПО PHOTOMOD.

Архитектура и особенности интеллектуальной информационной системы

Интеллектуальная информационная система представляет собой интегрированную платформу, включающую несколько функциональных модулей: модуль сбора полевых данных с использованием БПЛА, модуль наземной обработки ДЗЗ, ГИС-подсистему геопространственного анализа и интеллектуальное ядро, основанное на методах машинного и глубокого обучения.

Модуль сбора данных

Для получения высокоточных первичных данных использовались БПЛА, оснащенные навигационно-измерительными системами (GPS/RTK, IMU) и съемочными сенсорами (RGB- и мультиспектральными камерами). Применение RTK-позиционирования позволило снизить горизонтальную ошибку привязки до

2–5 см, что является критически важным для обеспечения точности последующих ГИС-аналитических операций.

Модуль обработки данных ДЗЗ

На этапе обработки включаются процедуры радиометрической и геометрической коррекции снимков, построение ортофотомозаики, генерация цифровой модели рельефа (ЦМР), расчет вегетационных индексов (NDVI, NDRE), а также формирование облаков точек с использованием методов Structure-from-Motion (SfM). Использование автоматизированных конвейеров обработки позволило значительно сократить время получения конечных продуктов.

ГИС-модуль пространственного анализа

ГИС-модуль обеспечивает интеграцию различной пространственной информации, включая результаты БПЛА-съемки, спутниковые данные, слои рельефа, почвенные карты и внешние статистические данные. Применяются методы пространственной интерполяции, overlay-анализа, кластеризации и моделирования зон риска.

Интеллектуальное ядро системы

Интеллектуальное ядро реализует алгоритмы глубокого обучения:

- сверточные нейронные сети (U-Net, DeepLabV3+) - для сегментации территорий;
- алгоритмы детекции объектов (YOLO, Mask R-CNN);
- модели временных рядов (LSTM, Transformer) - для прогнозирования динамики изменений;
- методы оптимизации маршрутов БПЛА (DQN, PPO).

Архитектура системы обеспечивает комплексный цикл «сбор - обработка - анализ - прогноз - управление», что отличает ее от традиционных систем, обладающих ограниченным уровнем автоматизации.

Методология исследования эффективности

1. Точностные характеристики: точность классификации (IoU, F1-score); ошибка геопривязки (RMSE); корректность пространственных моделей.
2. Производительность системы: время обработки данных (обработка ортофото, расчет индексов); скорость генерации прогнозных моделей.
3. Надежность и устойчивость: количество ошибок позиционирования; отказоустойчивость вычислительных модулей.
4. Энергетическая эффективность: расход заряда батареи БПЛА на миссию; влияние интеллектуальных алгоритмов планирования маршрутов на расход энергии.
5. Экономическая эффективность: сравнительные затраты на мониторинг; стоимость обработки 1 км² территории.

Экспериментальная часть

Экспериментальная оценка системы проводилась на данных полевых полетов БПЛА и спутниковых снимков Sentinel-2. Для сравнения использовались показатели традиционной обработки (ручная сегментация, стандартные ГИС-инструменты) и интеллектуальной обработки с применением нейросетевых моделей.

Интеграция нейросетевых моделей позволила повысить точность анализа более чем на 20–25 %.

Эффективность обработки данных. Время обработки 1 км² данных БПЛА:

- традиционный подход: 6–8 часов;
- интеллектуальный конвейер: 1–1.5 часа.

Применение алгоритма DQN при планировании маршрута позволило снизить энергопотребление БПЛА на 12–17 %.

Результаты показывают, что использование методов обработки данных и пространственного анализа повышает эффективность мониторинга территорий.

К основным результатам относятся:

1. Повышение точности классификации и сегментации объектов при использовании нейросетевых моделей.
2. Сокращение времени обработки данных за счёт автоматизации этапов анализа.
3. Снижение затрат на проведение мониторинга за счёт уменьшения числа повторных вылетов и ручной обработки.
4. Совместное использование данных БПЛА, ДЗЗ и ГИС обеспечивает более полное представление о состоянии территорий.

Проведенное исследование показало высокую эффективность реализации интеллектуальной информационной системы, основанной на интеграции средств БПЛА, методов ДЗЗ и геоинформационного анализа. Интеллектуальное ядро системы обеспечивает высокую точность выделения объектов, оперативность обработки данных и повышение общей продуктивности мониторинговых процессов. Результаты подтверждают целесообразность внедрения подобных систем в различные области: сельское хозяйство, экология, управление природными ресурсами, инфраструктурный мониторинг и др.

Компания «Ракурс» для тестирования получила более 20 блоков аэросъемки, выполненных различными организациями. К сожалению, часть предоставленных данных не содержала опорных точек или была получена неоткалиброванными камерами, что не позволяло оценить точность обработки. Далее рассмотрим в деталях результаты одного из таких блоков.

Съемка с откалиброванной камеры

- Количество маршрутов: 26
- Всего снимков: 595
- Камера: Canon EOS 500D (предварительно откалибрована)
- Высота полета: ~500 м

- GSD (размер пикселя на местности): ~8 см
- Опорные точки: 25 (точность до 10 см)
- Перепад высот: ~70 м

Первоначально блок был обработан в автоматическом режиме, без использования опорных точек и уравнивания. Привязка снимков производилась по центрам проекций, а ортотрансформирование осуществлялось в модуле PHOTOMOD GeoMosaic без учета рельефа. Контроль показал, что ошибки на опорных точках превышали 17 м, что связано с перепадом высот и неточной телеметрией.

Строгая фотограмметрическая обработка

При уравнивании блока 3 точки были выделены как контрольные.

Результаты по точностям:

- Ср. квадр. ошибка по опорным точкам: 15 см, 16 см, 12 см
- По контрольным точкам: 23 см, 29 см, 57 см
- По связующим точкам: 8 см, 14 см, 69 см

Общий вид блока представлен на следующем рисунке 54. Наибольшие ошибки на связующих точках наблюдаются на краях и в углах снимков.

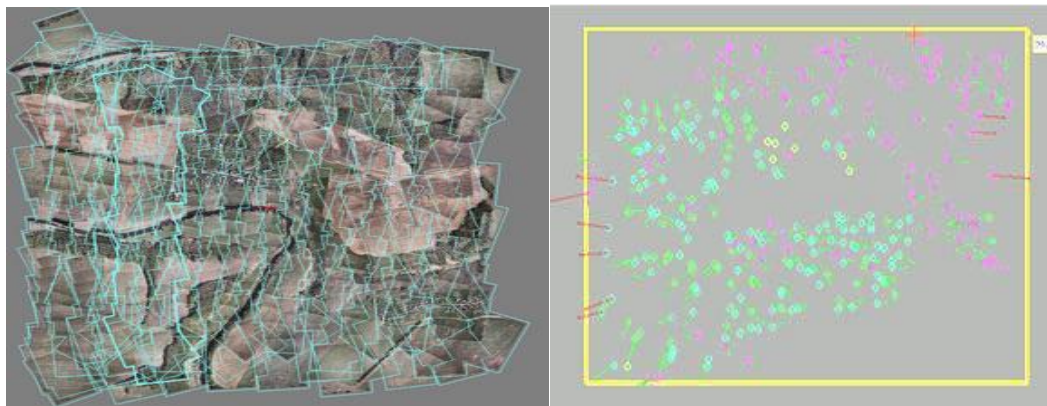


Рисунок 54 – Общий вид «блока» и величины ошибок на связующих точках

Анализ показал наличие систематической ошибки телеметрии по высоте Z, составляющей около 10,5 м. После ее устранения, ошибки на центрах проекций составили: 84 см (X), 239 см (Y), 75 см (Z). Большие отклонения по оси Y указывают на неточное определение времени съемки в телеметрии, а ошибки по Z - на проблемы калибровки камеры с щелевым затвором.

Построение ортофотоплана

На следующем этапе выполнено построение цифрового рельефа в автоматическом режиме и ортотрансформирование с его учетом. На рисунке 55 представлен фрагмент ортофотоплана без выравнивания яркости, демонстрирующий точность совмещения контуров соседних снимков.



Рисунок 55 – Фрагмент ортофотоплана без выравнивания яркости

Съемка открытого горнорудного месторождения в Центральном Казахстане

Объектом дальнейшего анализа стало горнорудное месторождение, расположенное в Центральном Казахстане, которое разрабатывается с 1985 года и имеет размеры около $2 \times 3 \times 0.25$ км. Среднегодовой объем добычи составляет 30 млн тонн, при удельной плотности $1,33 \text{ м}^3/\text{т}$. Проектная мощность по расчетам института «Карагандагипрошахт» - 20 млн тонн в год [25, 26].

Целью аэросъемки с БПЛА была трехмерная реконструкция и мониторинг карьера, включая визуализацию склонов и объемный анализ выемок породы. В работе использовались:

- БПЛА 101
- БПЛА собственной сборки

Оба аппарата оснащались цифровой камерой Sony NEX-5N с фокусным расстоянием 28 мм. В БПЛА 101 камера была закреплена жестко, а в опытном экземпляре использовалась двухосевая гиростабилизированная платформа, обеспечивающая постоянную ориентацию камеры (Рисунок 56).



Рисунок 56 – БПЛА

В таблице 20 приведена статистика полетов.

Таблица 20 – Статистика полетов

	БПЛА 101	Созданный опытный образец БПЛА
Количество вылетов	3	8
Количество фотографий	2435	824
Рабочие фотографии	1898	807
Высота от точки старта (м)	150	100
Точка старта	Поверхность земли	Внутри месторождения

Съемка проводилась с перекрытием 80%, что позволило устранить кадры с дефектами и смазанностью, а также исключить снимки, сделанные при разворотах (Рисунок 57).

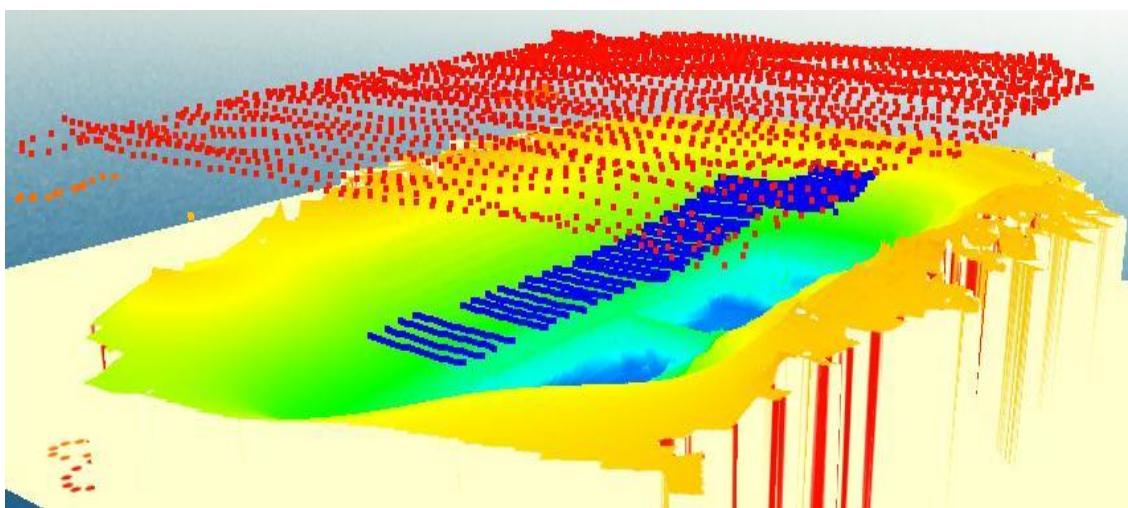


Рисунок 57 – Маршруты полетов БПЛА 101 (красные точки) и созданного опытного образца БПЛА (синие точки) над открытым горнорудным месторождением

Опытный БПЛА запускался из глубины порядка 200 метров ниже уровня поверхности. Он поднимался на 100 метров и выполнял съемку в противоположном направлении, что позволяло улучшить качество покрытия и обеспечить максимальную плотность точек съемки при меньшем числе маршрутов.

На рисунке 58 показан и сбор информации над открытым горнорудным месторождением с помощью БПЛА.



Рисунок 58 – Сбор информации над открытым горнорудным месторождением

Одной из особенностей аэросъемки карьеров с применением БПЛА является различие в высотах точек старта аппаратов, что может вызывать трудности при фотограмметрической обработке. Эти затруднения были устранены ручной расстановкой связующих точек на снимках, выполненных с разных высот. Сбор информации над открытым горнорудным месторождением (Рисунок 59).

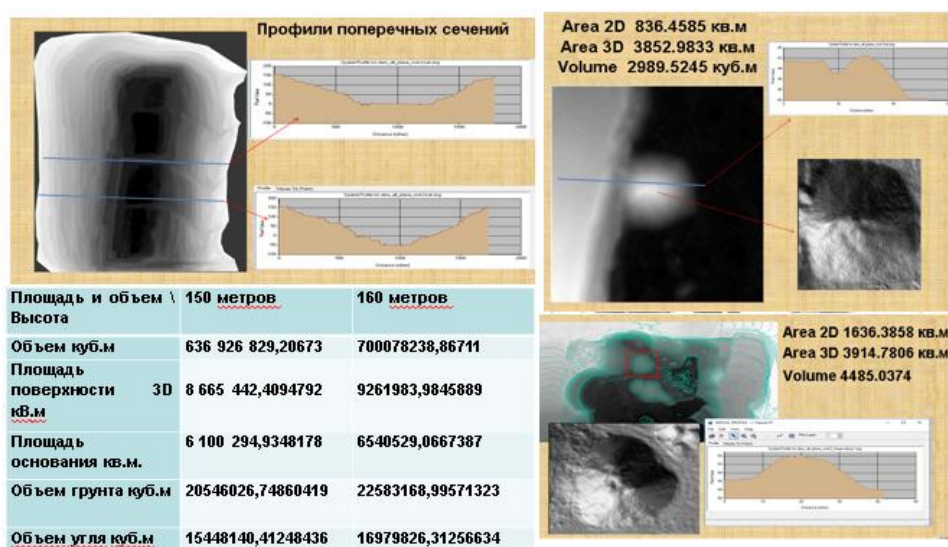


Рисунок 59 – Сбор информации над открытым горнорудным месторождением

Расстановка опорных точек и исходные данные. До начала съемки по всей площади месторождения была равномерно выполнена расстановка опорных знаков, координаты которых были замерены геодезическими методами. Расположение опорных точек показано на рисунке 60. В качестве знаков

использовались ленты из клеенчатого материала (ширина – 25 см, длина – 200 см), сложенные под углом 90°.

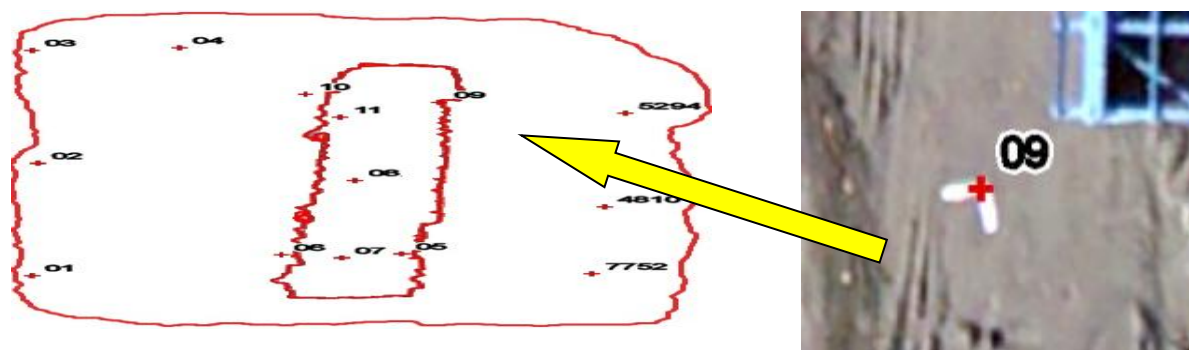


Рисунок 60 – Схема расстановки опорных точек

Фотограмметрическая обработка выполнена в среде Agisoft Metashape. Уравнивание координат осуществлялось итеративным методом до достижения минимальных отклонений по координатам. Для запуска очередной итерации применялись функции Ассерт и Run, а окончательное завершение процесса осуществлялось через диалоговое окно Adjust Coordinates.

В процессе обработки могли возникать ошибки, связанные с:

- превышением допустимого числа участков (более 5000);
- низкой связностью сетей точек;
- недостаточным количеством опорных точек на корректируемом участке.

Согласно нормативам, средняя погрешность положения пунктов съемочной сети не должна превышать 0,4 мм в масштабе карты по плану и 0,2 м по высоте (в масштабе 1:500 - соответствует 20 см на местности) [57].

Ошибки по опорным точкам после уравнивания приведены в таблице 21.

Таблица 21– Ошибки по опорным точкам после уравнивания

Имя точки	Ошибка (см)			
	X	Y	Z	Общая
03	1.42623	0.0190399	-0.629046	1.55891
02	-1.05392	1.79315	2.1179	2.96844
05	-2.19141	7.21445	-1.85466	7.76469
07	-7.40204	-3.55447	-0.388473	8.22055
08	-0.256165	0.873776	-2.55333	2.71083
09	-1.69852	-7.32364	-.0581117	7.54045
04	-6.31944	5.04837	3.02228	8.63456
06	5.36666	0.387021	-0.220922	5.38513
10	3.87104	-5.01862	0.855148	6.39553

Имя точки	Ошибка (см)			
	X	Y	Z	Общая
11	0.148656	-4.05701	-2.21976	4.62696
4810	4.0133	1.82096	2.67144	5.15355
5294	3.88597	2.28367	1.58826	4.77896
7752	0.49976	0.776777	0.629407	1.11772
Суммарная	3.73021	3.90971	1.75906	5.68284

Для анализа зависимости ошибок использован метод наименьших квадратов (таблица 22).

Таблица 22 – Уравнивание ошибок по опорным точкам методом наименьших квадратов

Имя точки	X	Y	Z	U	X*X	X*Y	X*Z	Y*Y	Z*Z	U*X	U*Y	U*Z	Y*Z
3	1,426	0,019	-0,929	1,559	2,034	0,027	-1,325	0,000	0,863	2,223	0,030	-1,448	-0,018
2	-1,054	1,793	-0,429	2,968	1,111	-1,890	0,452	3,215	0,184	-3,128	5,323	-1,274	-0,769
5	-2,191	7,214	-0,855	1,765	4,802	-15,810	1,873	52,048	0,730	-3,867	12,731	-1,508	-6,166
7	-7,402	-3,554	-0,388	0,221	54,790	26,310	2,875	12,634	0,151	-1,633	-0,784	-0,086	1,381
8	-0,256	0,874	-2,553	2,711	0,066	-0,224	0,654	0,763	6,519	-0,694	2,369	-6,922	-2,231
9	-1,699	-7,324	-0,058	1,540	2,885	12,439	0,099	53,636	0,003	-2,616	-11,282	-0,090	0,426
4	-6,319	5,048	3,022	1,635	39,935	-31,903	-19,099	25,486	9,134	-10,330	8,252	4,940	15,258
6	5,367	0,387	-0,221	5,385	28,801	2,077	-1,186	0,150	0,049	28,900	2,084	-1,190	-0,086
10	3,871	-5,019	0,855	2,396	14,985	-19,427	3,310	25,187	0,731	9,273	-12,022	2,049	-4,292
11	0,149	-4,057	0,255	0,627	0,022	-0,603	0,038	16,459	0,065	0,093	-2,544	0,160	-1,035
4810	4,013	1,821	0,655	0,154	16,107	7,308	2,629	3,316	0,429	0,616	0,280	0,101	1,193
5294	3,886	2,284	0,455	0,779	15,101	8,874	1,769	5,215	0,207	3,027	1,779	0,355	1,039
7752	0,500	0,777	0,629	1,118	0,250	0,388	0,315	0,603	0,396	0,559	0,868	0,704	0,489
Суммарная	0,290	0,263	0,439	22,856	180,888	-12,432	-7,596	198,714	19,463	22,423	7,084	-4,209	5,189

Согласно которому функция отклика (уровень смещения) определяется как (30):

$$\sum U_i = A_1 * \sum X_i + A_2 * \sum Y_i + A_3 * \sum Z_i \quad (30)$$

Максимальное влияние на функцию оказывает компонент Y, что выражается следующей моделью (31):

$$U = 0,12 * X + Y - 0,07 * Z \quad (31)$$

На основе этих данных построена модель, показанная на рисунке 61, где полученная модель представляет цифровую модель отвала.

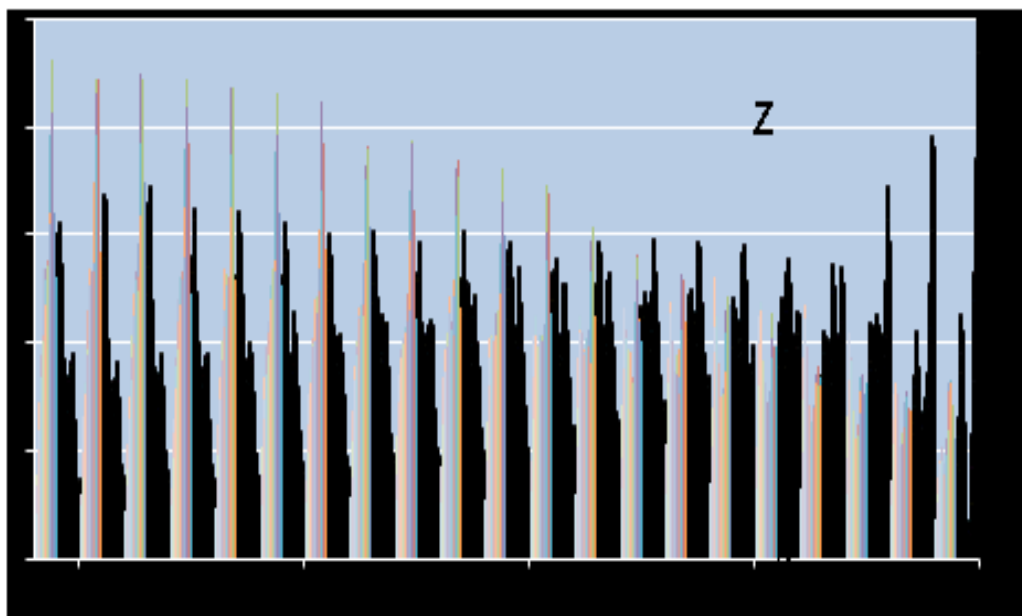


Рисунок 61 – Цифровая модель отвала

Обработка данных в Agisoft Metashape велась в распределенной вычислительной среде (5 ПК), где один из компьютеров использовался как сервер.:

- Найдено: 993 518 связующих точек
- Построено плотное облако: ~761 млн точек
- Разрешение на местности: 12.5 см/пикс
- Плотность точек: 64 точек/м²

Цифровая модель местности и поперечные профили (Рисунок 62).

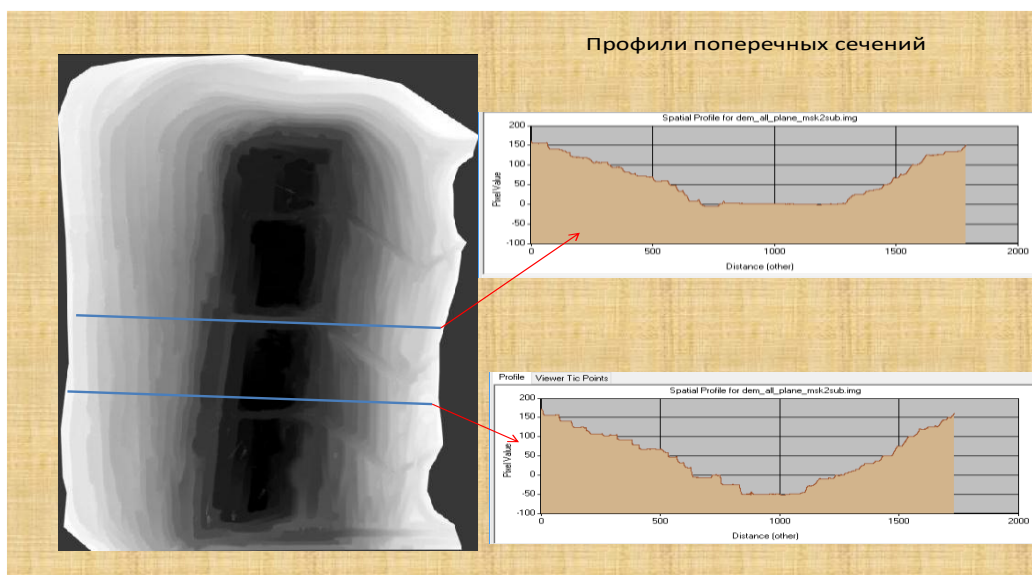


Рисунок 62 – Цифровая модель местности и поперечные профили

На основе модели выполнена оценка объемов вынутого грунта при разных высотах отсчета (таблица 23).

Таблица 23 – Объем вынутого грунта в зависимости от выбора начальной высоты

Площадь и объем \ Высота	150 метров	160 метров
Объем куб.м	636 926 829,20673	700 078 238,86711
Площадь поверхности 3D кВ.м	8 665 442,4094792	9 261 983,9845889
Площадь основания кв.м.	6 100 294,9348178	6 540 529,0667387
Объем грунта куб.м	20 546 026,74860419	22 583 168,99571323
Объем угля куб.м	15 448 140,41248436	16 979 826,31256634

Полученная ЦММ позволила вычислить объемы внутренних производственных насыпей, что показано на рисунке 63.

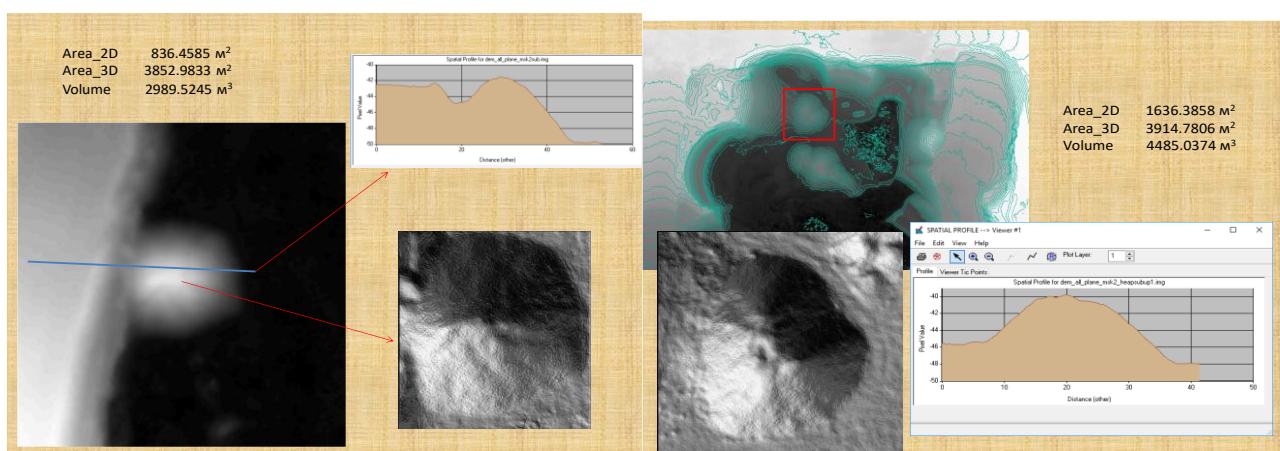


Рисунок 63 – Объем первой внутренней производственной насыпи и объем второй насыпи

С учетом начала эксплуатации месторождения в 1985 году, средняя годовая производительность составляет ~21 млн м³. Применен коэффициент пересчета грунта в руду 1.33, принятый для данного месторождения.

В таблице 24 показаны затраты машинного времени на обработку данных с БПЛА.

Таблица 24 – Затраты на обработку данных с БПЛА

№	Операция	Компьютеры	Время
1	Загрузка фотоснимков	1	1 - 5 мин.
2	Поиск связующих точек	4	3 ч.
3	Построение плотного облака	4	37 ч. 18 мин.

4	Построение модели	1	1 ч. 37 мин.
5	Построение текстуры	1	6 ч. 55 мин.
6	Построение тайловой модели	4	8 ч. 52 мин.
7	Построение карты высот	4	14 мин
8	Создание ортофотоплана	1	55 мин
9	Построение горизонталей	1	30 мин.

Общее время всех этапов обработки составило 128 часов (~20 рабочих дней). В результате проведенных исследований подтверждена высокая эффективность применения БПЛА для аэросъемки открытых горнорудных месторождений. Методика обеспечивает:

- получение детализированных ортофотопланов;
- построение цифровых моделей рельефа и отвалов;
- оценку объемов выработки;
- возможность оперативного мониторинга производственных процессов.

Рекомендуется использовать БПЛА самолетного типа (БПЛА 101) для съемки всего месторождения, а вертолетные платформы – для внутренних и узких объектов.

Также подчеркивается, что достижение высокой точности возможно только при соблюдении следующих условий:

- корректная организация съемочного процесса и соблюдение норм перекрытия;
- использование откалиброванных фотокамер;
- строгая фотограмметрическая обработка.

Ортофотоплан участка железнодорожного полотна. Высота съемки 100 м. Разрешение 5 с. (Рисунок 64).



Рисунок 64 - Ортофотоплан участка железнодорожного полотна. Высота съемки 100 м. Разрешение 5 см

Использование БПЛА в горнорудной промышленности демонстрирует огромный потенциал для автоматизации картографических работ, сокращения сроков полевого этапа и повышения достоверности данных.

4.2 Интеллектуальная технология получения максимальной точности результатов съемки

Сбор данных сопровождался фиксацией координат точек съемки с использованием двухчастотного GNSS-приемника в режиме RTK. Привязка снимков к местности осуществлялась по наземным контрольным точкам, координаты которых были получены с использованием геодезических приемников с точностью до 1 см. Все снимки обрабатывались в программном обеспечении Agisoft Metashape с формированием облаков точек, цифровой модели рельефа и ортофотоплана.

Для достижения высокой точности и достоверности фотограмметрических продуктов, получаемых с использованием БПЛА, разработан ряд рекомендаций, адресованных как пользователям, так и разработчикам беспилотных летательных платформ. Эти рекомендации основаны на практических испытаниях и анализе множества аэрофотосъемочных миссий.

Технические рекомендации

- Использовать предварительно откалиброванные фотокамеры.
- Ограничивать выдержку съемки не более $1/250$ с, чтобы минимизировать смазывание.

- Применять объективы с фиксированным фокусным расстоянием; при использовании зум-объектива следует жестко зафиксировать масштаб.
- Осуществлять съемку с фокусировкой на бесконечность при выключенном автофокусе.
- Обеспечивать перекрытие снимков: 80% продольное и 40–60% поперечное.
- Предпочтительно использовать камеры с центральным затвором.
- Применять двухчастотные GPS-приемники и, по возможности, дифференциальную коррекцию (RTK/PPK).
- Устанавливать на борту IMU-модули, даже если они обладают умеренной точностью.

Тестовые полигоны и условия отбора

Для проведения испытаний опытного образца БПЛА был сформулирован ряд требований к полигонам:

- доступность и безопасность полетов;
- удаленность от аэропортов (не ближе 5 км);
- расположение вне густонаселенных зон;
- возможность установки опорных знаков;
- площадь полигона - не более 1 км².

В результате отбора были определены два тестовых полигона:

- Акватория озера Сайран - в засушливый период (без заполнения водой);
- Участок в районе пос. Алатау, заменивший более удаленный Байсерке.

Аэросъемка озера «Сайран»

Планирование маршрута съемки было выполнено с учетом оптимизации времени (таблиц 25), перекрытий и эффективности маршрутов (рисунок (65)).

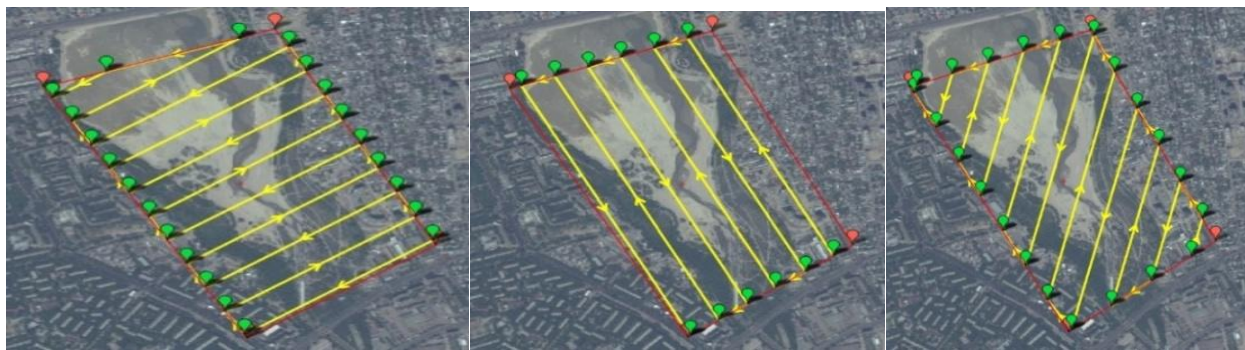


Рисунок 65 – Схемы маршрутов для выполнения съемки

Таблица 25 – Оптимальный план полета и съемки участка «Сайран»

№ мар	Высо та	Продоль ное	Попереч ное	Фотоаппа рат	Фок ус	Кол- во	Вре мя	Приме чание
----------	------------	----------------	----------------	-----------------	-----------	------------	-----------	----------------

ш- рута	полет а	перекрыт ие	перекрыт ие			снимк ов		
1	200 м	80 %	60 %	Nex5N	18 мм	149	35,1 4 мин	Оптим. по времен и
2	200 м	80 %	60 %	Nex5N	18 мм	153	39,4 9 мин	
3	200 м	80 %	60 %	Nex5N	18 мм	147	39,5 6 мин	

Съемка проводилась на высоте 200 м, с фокусным расстоянием 18 мм, камерами Sony NEX-5N. Были получены 168 снимков с GPS-данными, отражающими координаты, скорость и параметры ориентирования (рисунок 66).

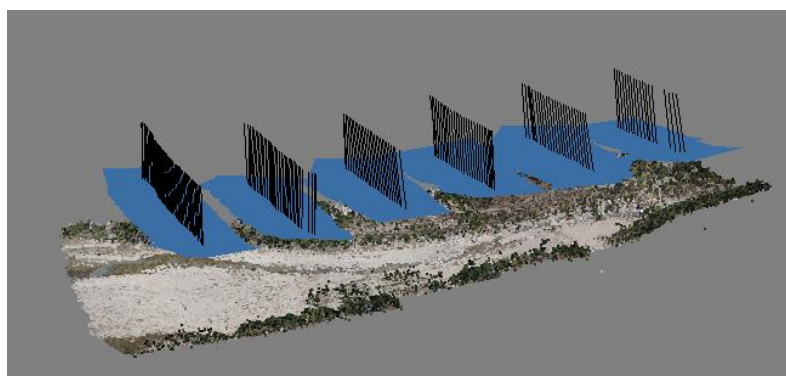


Рисунок 66 – Организация полетов и съемки

Полученные снимки имели разрешение 5 см/пикс. Для повышения точности применялись опорные наземные точки, координаты которых были определены с использованием высокоточных GPS-приемников (погрешность не более 16 см). Пример размещения таких точек показан на рисунке 67, а их координаты - в таблице 26.



Рисунок 67 – Распределение опорных точек по исследуемой территории и векторная карта озера «Сайран»

Таблица 26 – Координаты опорных точек

Точка	X (м)	Y (м)	Z (м)
91	651697.809	4788859.831	777
92	651850.549	4788750.282	779
95	651930.435	4788750.218	786
101	651400.150	4789202.690	777
102	651175.300	4789394.990	772
103	651813.210	4789279.180	793

Обработка производилась с применением ПО PHOTOMOD и PhotoScan (Agisoft Metashape). На всех снимках были вручную расставлены связующие точки, выполнено уравнивание, построена триангуляционная сеть.

На рисунке 68 показан ортофотоплан отснятой территории и многослойная векторная карта территории озера «Сайран».



Рисунок 68 – Ортофотоплан территории озера «Сайран»

Аэросъемка поселка «Алатау». Участок был выбран в связи с началом строительства инфраструктурных объектов, включая дороги и здания научного комплекса (рисунок 69).

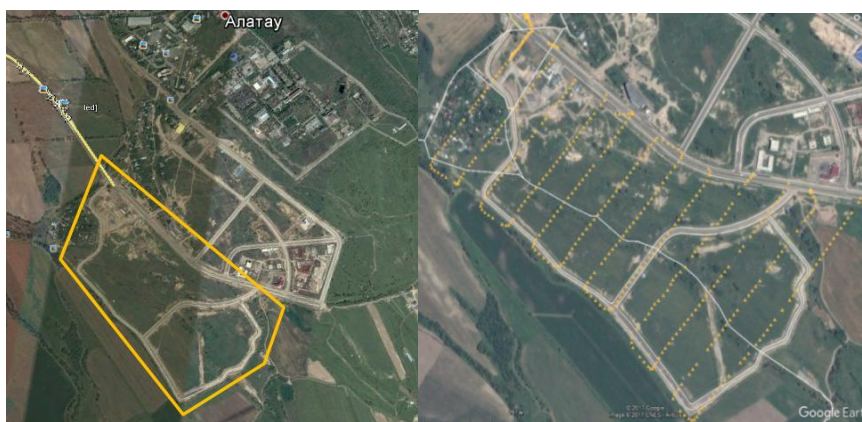


Рисунок 69 – Планируемая территория съемки

План съемки включал точки маршрута и места срабатывания камеры, как показано на рисунке 70.

Для геопривязки снимков были расставлены контрольные наземные точки в конфигурации «конверт»: по углам и в центре исследуемой зоны (рисунок 74). Контрольные метки представляли собой кресты из контрастного материала (рисунок 92).

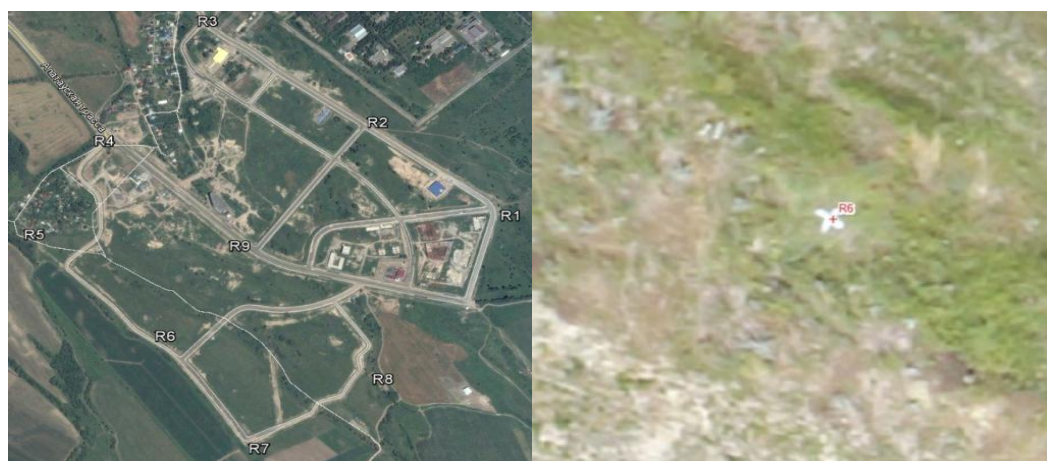


Рисунок 70 – Расположение контрольных точек на полигоне. Наземная контрольная точка

Результатом обработки стал ортофотоплан в масштабе 1:500, пример которого приведен на рисунке 71.



Рисунок 71 – Часть ортофотоплана в масштабе 1:50

Разработка технологии преобразования исходных цифровых картографических данных в формат программного обеспечения ГИС.

ГИС – это инструменты, предназначенные для хранения, анализа и визуализации пространственной информации, связанной с различными объектами местности. В контексте данного исследования ГИС используются для обработки и последующей визуализации данных, полученных в результате аэрофотосъемки, включая ортофотопланы, координаты привязки и центры проекций.

Географическая привязка аэроснимков

Первоначальным этапом интеграции данных в ГИС является координатная привязка фотоматериалов, полученных в ходе аэрофотосъемки. Эта операция обеспечивает соответствие между реальными географическими координатами и пикселями на снимках, а также позволяет обогатить изображения семантической информацией ("река", "здание", "дорога").

Обработка центров проекций

Координаты центров фотографирования, извлеченные из IMU или GPS-логов полетных контроллеров (в формате .tel, .csv, .txt и др.), обеспечивают автоматизированную привязку снимков. Для этой цели была разработана специализированная программа CAMERA, позволяющая извлекать необходимые данные из лог-файлов (рисунок 72).

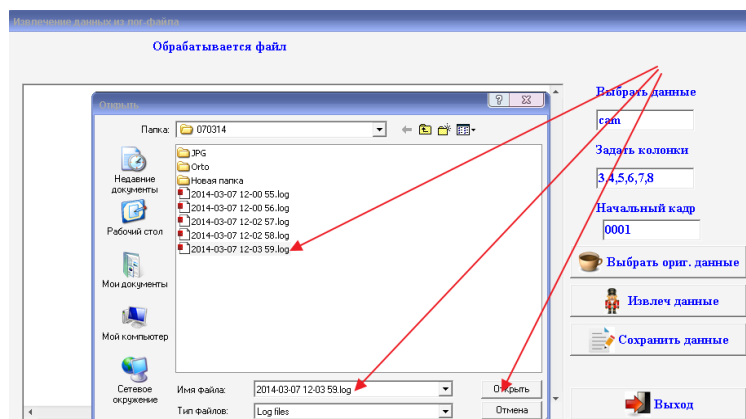


Рисунок 72 – Оконный интерфейс программы «CAMERA»
(выбор лог-файла)

Основные этапы преобразования данных в формат ГИС

1. Формирование ортофотоплана из отдельных аэрофотоснимков.
2. Привязка ортофотоплана к выбранной системе координат (WGS84 / UTM zone 43N).
3. Классификация объектов на изображениях по типам (гидрография, растительность, строения и т.д.).
4. Векторизация объектов - превращение растровых данных в цифровые векторные формы.
5. Создание атрибутивной базы данных, содержащей характеристики объектов.
6. Связывание векторных объектов с таблицей атрибутов.
7. Заполнение полей таблицы на основе семантики объектов.
8. Настройка системы запросов к базе данных, обеспечивающей выборку по интересующим объектам.

ГИС водохранилища «Сайран»

Система была создана в координатной системе WGS84 / UTM zone 43N. В результате векторизации были выделены следующие категории объектов:

- Гидрография: впадающие реки.
- Растительность: деревья, кустарники.
- Строения: жилые, складские, офисные помещения, беседки, контейнеры.
- Дорожная сеть: асфальтированные дороги.
- Рельеф: изолинии высот.

На рисунке 73 показаны примеры участков ортофотоплана с распознанными растительными объектами.



а) фрагмент ортофотоплана с кустарниками



б) фрагмент ортофотоплана с деревьями

Рисунок 73 – Кустарники и деревья

Ортотрансформирование и создание цифровой модели рельефа (ЦМР)

После первичного уравнивания фотоснимков формируется облако точек, на основе которого создается матрица высот - цифровая модель рельефа. Затем возможно построение текстурированной 3D модели, используя:

- векторную карту;
- цифровые фотоснимки;
- триангуляционные модели (TIN);
- базы фасадов (в CityEngine).

Построение 3D-моделей: программные инструменты

Используемые программы: AutoCAD - 2D/3D проектирование, 3D MAX - Визуализация, анимация, CityEngine - 3D-моделирование городской среды на основе геоданных, ContextCapture - 3D-моделирование из фотографий, включая съемки с БПЛА, Agisoft Metashape - Создание ЦМР, ортофотопланов, 3D моделей, GIS Sputnik - Визуализация и анализ данных с БПЛА (рисунок 74).



Рисунок 74 – Ортофотоплан участка с дачными домиками

Этапы построения 3D-моделей (Metashape/ContextCapture)

1. Импорт фотографий и координат центров проекции.
2. Построение разреженного облака точек.
3. Формирование плотного облака.
4. Построение модели поверхности.
5. Создание текстуры.
6. Формирование тайловой 3D-модели.

Создание ЦМР и ортофотоплана - по требованию. *Пример 3D-визуализации в ESRI CityEngine*

Исходные векторные данные, подготовленные в ArcGIS, были импортированы в CityEngine для создания трехмерной застройки поселка «Алатау». Высоты зданий и фасадные формы были заданы вручную на основе аэросъемки с БПЛА (рисунок 75).

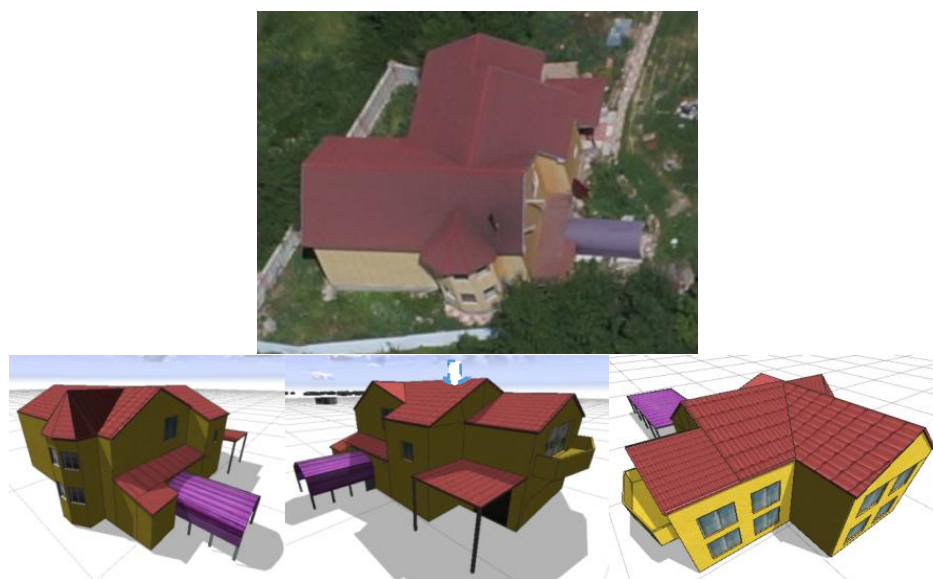


Рисунок 75 – Построение фасада с использованием CGA и БПЛА-изображений

Модель создавалась с использованием языка CGA (Computer Generated Architecture) с поддержкой скриптов на Python. Это позволило не только автоматизировать генерацию фасадов, но и редактировать отдельные элементы визуализации.

Точность 3D моделей и возможность измерений

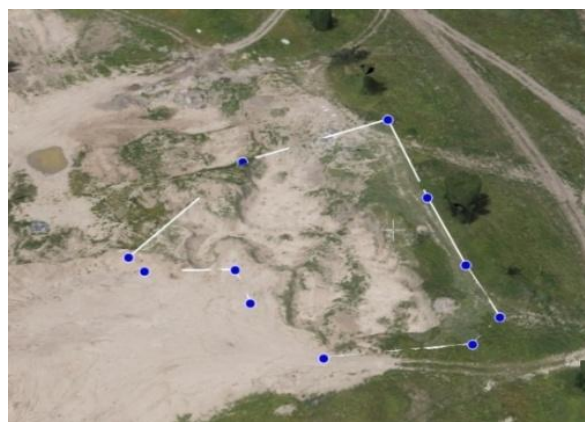
Полученная 3D модель представляет реальное состояние застройки, на которой можно проводить замеры и анализ (рисунки 76, 77).



Рисунок 76 – 3D-модель того же участка, вид сбоку



а) участок для выполнения измерений на ортофотоплане



б) участок для измерений

Рисунок 77 – Участок для выполнения измерений

Модель позволяет оценить:

- объемы выемок и насыпей;
- геометрию объектов (высота, ширина, этажность);
- планировку и рельеф территории.

Пример из ContextCapture (рисунок 78).



а) участок для сравнения



б) участок для измерений

Рисунок 78 – 3D-модель участка

Сравнение программ обработки

ESRI CityEngine. Если есть векторные данные с подробной атрибутивной информацией, то можно достаточно быстро создать 3D-модель поселка с типовой застройкой. В этой модели можно проводить только простейшие расчеты - расстояния, площадь.

Agisoft Metashape и *ContextCapture*. Обе программы имеют примерно одинаковый функционал и позволяют построить реальную 3D-модель поселка. На этой модели уже можно проводить 3D измерения - измерять объемы.

4.3 Исследование эффективности применения алгоритмов анализа изображений в задаче навигации БПЛА

Для верификации полученных результатов навигации проведено сравнение данных, полученных с использованием предложенного подхода, с эталонными траекториями, измеренными с помощью высокоточного GNSS-оборудования. Средняя ошибка навигации по предложенному методу составила менее 3.2 м, что превышает точностные характеристики типовых визуальных навигационных систем.

Аппаратные и методологические основы навигации

Навигационные системы играют ключевую роль в реализации автоматических и автоматизированных режимов полета БПЛА. В современном поколении навигационного оборудования, как правило, используется схема комплексирования спутниковых и инерциальных навигационных систем, позволяющая достичь высокой точности позиционирования при оптимальных массогабаритных характеристиках.

Тем не менее, спутниковые навигационные системы имеют существенные ограничения, главным из которых является ухудшение точности позиционирования в условиях плохого приема сигнала. Инерциальные системы, в свою очередь, накапливают ошибки без коррекции. Альтернативным источником навигационных данных может служить видеосъемка с борта БПЛА. Вопросы использования видеоданных для навигации БПЛА подробно исследованы в ряде научных работ [28].

Сопоставление изображений и использование особых точек

Применяемые методы визуальной навигации делятся на:

- методы, использующие сопоставление с эталонными изображениями (спутниковыми/аэрофотоснимками);
- методы оптического счисления пути, использующие последовательные кадры видеопотока.

Обе группы требуют решения задачи сопоставления изображений, с учетом искажений ракурса, масштаба и других факторов. Для повышения устойчивости применяются методы, использующие особые точки и их дескрипторы - компактные числовые описания яркостных или геометрических признаков в окрестности точки.

Алгоритмы анализа изображений: SIFT и ORB

Цель данного исследования - оценка эффективности алгоритмов SIFT [139] и ORB [45] в задаче навигации БПЛА.

На эталонных изображениях местности размечаются опорные участки (ОУ). Каждому ОУ соответствуют известные координаты, размеры и ориентация. Во время полета из текущего изображения (ТИ) выделяются особые точки и сопоставляются с базой эталонных точек (БОТ). При совпадении достаточного

числа точек (по порогу) подтверждается наличие ОУ в текущем кадре, и определяется его положение и масштаб. Для повышения точности используется метод RANSAC [46] и фильтрация Калмана.

Реализация и настройка алгоритма SIFT

Алгоритм SIFT применялся для детектирования устойчивых ключевых точек и построения дескрипторов в задачах распознавания объектов и построения карты местности. Реализация выполнялась на основе библиотеки OpenCV (версия 4.x) с использованием следующей конфигурации:

1. Построение гауссовой пирамиды - 4 октавы, 5 уровней в октаве.
2. Поиск экстремумов DOG-функции - порог контрастности: 0.04, фильтрация по краевым откликам: $r = 10$
3. Определение ориентации ключевой точки - окно поиска 16×16
4. Формирование дескриптора- размерность 128, гауссово взвешивание с $\sigma = 8$

Цели использования

- привязка визуальных точек при построении траектории UAV,
- поиск соответствий между последовательными кадрами,
- построение разреженной 3D-реконструкции участка полета.

SIFT не требует обучения, однако требует оптимизации параметров для стабильности работы в условиях UAV: компенсирована дрожь камеры, автоматизирована нормализация контраста [47].

Реализация ORB (Oriented FAST и BRIEF). Поскольку SIFT обладает высокой вычислительной сложностью, для задач реального времени использован ORB - быстрый бинарный дескриптор.

Основные параметры реализации

- количество ключевых точек: 2000 на кадр,
- scale factor = 1.2,
- уровни пирамиды = 8,
- patch size = 31,
- BRIEF-дескриптор длиной 256 бит.

Улучшения в работе системы

Для повышения надежности сопоставления ORB были использованы:

- кросс-валидация матчей (cross-check),
- отбор по Hamming distance (< 30),
- RANSAC для исключения ложных соответствий.

ORB обеспечил работу системы в режиме 30–45 FPS, что позволило использовать алгоритм в подсистемах навигации UAV и для построения SLAM-карты.

Реализация SLAM и архитектура системы

Для решения задачи позиционирования и одновременного построения карты (SLAM) применялась модифицированная архитектура на основе ORB-SLAM2.

Структура SLAM-системы.

Система состоит из трех потоков:

(1) Tracking: извлечение ORB-ключевых точек, сопоставление с картой, оценка положения камеры методом PnP и RANSAC, фильтрация выбросов.

(2) Local Mapping: создание локальных ключевых кадров, триангуляция 3D-точек, локальная оптимизация BA (Bundle Adjustment).

(3) Loop Closing: поиск замыканий (Bag-of-Words модель), глобальная оптимизация карты, обновление графа позиций.

Используемые нейросетевые расширения

Для повышения точности использовались: CNN-модель для маскинга подвижных объектов (MobileNetV3), небольшой автоэнкодер для подавления шума и стабилизации изображения, DNN-модуль для прогнозирования оптического потока (liteFlowNet).

Нейросетевые компоненты обучались на: TUM RGB-D, EuRoC MAV, собственных UAV-данных (больше 8000 изображений).

Архитектура обучающихся моделей

Autoencoder для стабилизации изображений

- Encoder: 4 сверточных блока (Conv, ReLU и BatchNorm)
- Bottleneck: Dense(256)
- Decoder: ConvTranspose блоки
- loss: MSE

Функция: уменьшать шум, смазывание и вибрации камеры. CNN-модель для сегментации движущихся объектов

Использовалась мобильная модель: MobileNetV3-small backbone сегментационная голова DeepLabV3

- loss: BCE и Dice

Функция: убирать движущие объекты из карты, улучшая SLAM. Модель оптического потока

Использовалась облегченная версия FlowNet-Lite: две входные карты (I_t , I_{t+1}),

- сверточная U-Net архитектура (7 уровней), loss: Charbonnier loss.

Функция: предсказание движения камеры между кадрами при отсутствии четких ключевых точек.

Алгоритм SIFT [39]

Ключевые особенности: Построение масштабной гауссовой пирамиды [42]; Детектирование особых точек как экстремумов разностных изображений; Квадратичная интерполяция положения точки и масштаба; Отбрасывание точек на границах и шумовых экстремумов; Привязка к

направлению градиента (гистограмма ориентаций); Формирование дескриптора - вектор, устойчивый к повороту и масштабированию; Сравнение точек по Евклидовой норме между дескрипторами.

Алгоритм ORB [14]

ORB - упрощенная и ускоренная альтернатива: Детектирование точек с помощью алгоритма FAST [43]; Определение ориентации на основе центра масс яркостной окрестности; Формирование бинарных дескрипторов: сравнение яркостей в парах точек; Минимальные требования к ресурсам, высокая скорость вычислений; Хорошая устойчивость к простым геометрическим искажениям.

Программный комплекс для экспериментов [44]

Для проведения экспериментов был разработан программный комплекс, включающий: Имитацию видеосъемки в FlightGear - авиационном симуляторе с открытым кодом; Сетевой интерфейс управления виртуальной камерой, симуляцией условий наблюдения; Алгоритмы визуальной навигации с реализацией SIFT и ORB; Модули оценки эффективности и визуализации траекторий; Инструменты перехода к управлению дроном на основе визуальных данных.

Преимущества использования FlightGear: Отсутствие необходимости вручную моделировать 3D-ландшафт; Возможность задавать освещение, облачность и другие погодные параметры; Интеграция с траекторным управлением БПЛА.

В структуре разработанного программного комплекса информационные потоки представлены двумя типами: основные - сплошными стрелками, и вспомогательные - пунктирными. Все ключевые сведения, включая эталонные изображения, параметры траекторий, координаты опорных участков, а также результаты виртуального моделирования, агрегируются в единой централизованной базе данных.

Модуль управления изображениями предоставляет функционал как для генерации имитационных снимков с заданными характеристиками в симуляторе, так и для импорта реальных фотоснимков с дальнейшей интеграцией в БД. Маршруты полета задаются через визуальный интерфейс в виде последовательности опорных точек, каждая из которых характеризуется пространственными координатами, высотной меткой и временными параметрами. Расчет промежуточных узлов выполняется с применением сплайн-интерполяции [45].

Каждый опорный участок ассоциирован с определенным снимком, что позволяет их параметризовать и размечать в пределах комплекса. В рамках моделирования возможно задание атмосферных условий (сезон, освещение, облачность) и запись видеопотока виртуального полета, либо - выполнение оценки навигационных характеристик в реальном времени. Алгоритмы навигационного анализа реализованы как отдельная библиотека,

поддерживающая методы SIFT и ORB, используя библиотеку OpenCV с открытым исходным кодом. Полученные данные фиксируются в БД и доступны для статистического анализа через встроенный модуль визуализации.

Функциональные возможности включают:

- сравнение траектории полета с эталонной;
- визуализацию ошибок навигации;
- анализ эффективности применения методов компьютерного зрения в различных сценариях.

Рисунок 79 демонстрирует участки, покрытые кустарниковой растительностью и отдельные древесные объекты.



Рисунок 79 – Кустарники на местности и деревья на местности

Рисунок 80 отображает структуру дорожной сети в пределах искусственного водоема "Сайран".



Рисунок 80 – Дороги на местности

Рисунок 81 иллюстрирует элементы застройки в непосредственной близости от водохранилища, а Рисунок 82 - особенности рельефа исследуемой территории.



Рисунок 81 – Строения на местности

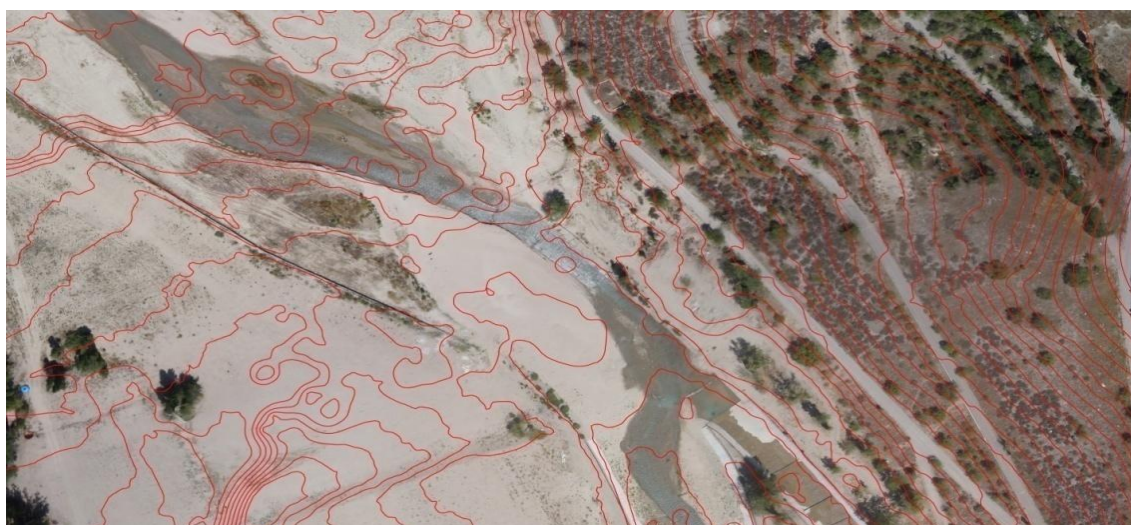


Рисунок 82 – Рельеф местности

Рисунок 106 представляет результат интеграции всех вышеуказанных данных в виде ИИС на базе БПЛА.

Созданная ГИС включает: геопривязанные ортофотопланы; цифровую карту местности; атрибутивные данные по слоям (растительность, строения, дороги, рельеф).

Система обеспечивает получение и обработку растровых и векторных данных на основе аэросъёмки с БПЛА. Это позволяет выполнять съёмку территорий, где применение космических или авиационных средств затруднено.

Рельеф рассматривается как сложный объект, параметры которого изменяются под воздействием природных и антропогенных факторов.

Традиционные методы моделирования, основанные на интерполяции или физических зависимостях, ограничены при работе с разреженными или неоднородными данными.

В связи с этим требуется применение методов, способных учитывать пространственные и временные изменения рельефа при ограниченном объеме исходной информации.

Использование БПЛА сокращает время сбора данных и уменьшает затраты на проведение мониторинга.

С развитием БПЛА, систем ДЗЗ и ГИС открываются новые возможности для получения высокоточных данных о рельефе в реальном времени. В сочетании с методами искусственного интеллекта, такими как нейросетевое моделирование, машинное обучение и предиктивная аналитика, становится возможным интеллектуальное прогнозирование рельефа, обеспечивающее высокую адаптивность к пространственно-временной изменчивости данных [46].

Рельеф как сложный объект моделирования

Рельеф характеризуется сложной структурой и многоуровневой природной и антропогенной обусловленностью. Он может быть представлен как динамическая система с нелинейными взаимодействиями между составляющими - высотами, уклонами, экспозицией, водоразделами и эрозионными структурами. В контексте интеллектуального моделирования рельеф трактуется как гетерогенная, пространственно распределенная система, чье поведение можно аппроксимировать с помощью обучающихся моделей на основе многоканальных данных.

К числу ключевых особенностей рельефа как сложного объекта относятся:

- Высокая размерность входных данных (изображения, LiDAR-облака точек, DEM, многоспектральные данные);
- Неоднородность и фрактальность пространственной структуры;
- Нелинейные связи между параметрами (влажность, уклон, растительность и др.);
- Темпоральные изменения, обусловленные сезонностью, эрозией и хозяйственной деятельностью.

Роль БПЛА и ГИС ДЗЗ в интеллектуальном прогнозировании

БПЛА оснащенные RGB, мультиспектральными или LiDAR-сенсорами обеспечивают детальное зондирование поверхности с высоким пространственным разрешением (до сантиметров). Это позволяет формировать цифровые модели местности (ЦММ), цифровые модели рельефа (ЦМР), а также собирать временные ряды изменений ландшафта. Данные, полученные с БПЛА, интегрируются в ГИС-платформы, где совмещаются с архивными спутниковыми данными и открытыми геоданными (SRTM, Sentinel-2 и др.).

Интеллектуальное прогнозирование в этом контексте предполагает:

- Создание обучающих выборок на основе пространственных данных (DEM, ортофотопланы, текстуры); Построение нейросетевых моделей (LSTM, CNN, GNN) для предсказания морфометрических параметров; Генерация сценариев изменения рельефа при различных внешних воздействиях (дожди, вырубка, строительство).

Методология прогнозирования

Метод прогнозирования рельефа включает следующие этапы:

1. Сбор и подготовка данных: ортотрансформированные снимки БПЛА, DEM, вегетационные индексы, климатические параметры. Выполняются очистка, нормализация и синхронизация данных.
2. Формирование признаков: расчёт высотных профилей, уклонов, кривизны и их агрегация в обучающие выборки.
3. Обучение модели: применение CNN для извлечения пространственных признаков и LSTM для учёта временной динамики.
4. Прогнозирование: построение прогнозных моделей рельефа.
5. Валидация: сравнение результатов с контрольными измерениями (GPS, повторные съёмки).

На этапе моделирования данные используются для обучения моделей, описывающих зависимость между входными параметрами и прогнозируемыми значениями.

Классические методы (регрессия, деревья решений) применяются для базовой оценки.

Нейросетевые модели (CNN, LSTM) используются для анализа пространственно-временных зависимостей и прогнозирования динамики изменений.

Обученные модели используются для:

- прогнозирования траектории движения БПЛА;
- оценки состояния полёта;
- анализа изменений по данным ДЗЗ.

Это позволяет учитывать неопределённость среды и поддерживать принятие решений при управлении.

Исходными данными для прогнозирования являются временные ряды навигационных параметров БПЛА (координаты, скорость, курс, высота), данные бортовых датчиков, а также информация, получаемая от систем ДЗЗ. На основе этих данных формируются обучающие выборки, отражающие динамику движения летательного аппарата.

Алгоритмы машинного обучения используются для аппроксимации зависимости между предыдущими состояниями БПЛА и его будущим положением. В представленном примере реализована авторегрессионная прогнозная модель, которая по нескольким предыдущим точкам траектории прогнозирует следующую позицию БПЛА. Подобные модели являются базой для

более сложных нейросетевых архитектур (LSTM, GRU), применяемых в интеллектуальных системах управления.

Полученные прогнозы используются: для предсказания траектории полета БПЛА; для коррекции маршрута в реальном времени; для повышения устойчивости управления; для интеграции данных ДЗЗ при мониторинге объектов и территорий [48]. Съёмки разными сенсорами показан на рисунке 83.



Рисунок 83 – Съёмки разными сенсорами

```
# Intelligent UAV control & remote sensing example
# Trajectory prediction using autoregressive ML model (NumPy)
import numpy as np
import matplotlib.pyplot as plt
# Generate synthetic UAV trajectory (x, y)
t = np.linspace(0, 20, 300)
x = np.sin(t)
y = np.cos(t)
data = np.vstack((x, y)).T
# Create autoregressive dataset
def create_ar(data, lag=5):
    X, Y = [], []
    for i in range(len(data) - lag):
        X.append(data[i:i+lag].flatten())
        Y.append(data[i+lag])
    return np.array(X), np.array(Y)
lag = 5
```

```

X, Y = create_ar(data, lag)
# Train/test split
split = int(0.8 * len(X))
X_train, X_test = X[:split], X[split:]
Y_train, Y_test = Y[:split], Y[split:]
# Train linear regression (least squares)
W, _, _, _ = np.linalg.lstsq(X_train, Y_train, rcond=None)
# Prediction
Y_pred = X_test @ W
# Plot true vs predicted trajectory
plt.figure()
plt.plot(Y_test[:,0], Y_test[:,1])
plt.plot(Y_pred[:,0], Y_pred[:,1])
plt.title("UAV Trajectory Prediction (ML autoregressive model)")
plt.xlabel("X")
plt.ylabel("Y")
plt.show()

```

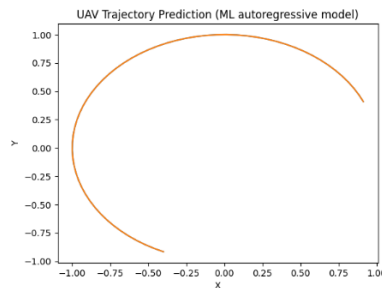


Рисунок 84 – ML autoregressive model

В рамках экспериментального участка была выполнена съемка с помощью БПЛА, после чего построены цифровые модели рельефа. На основе последовательностей DEM и многоспектральных индексов была обучена нейросеть LSTM для предсказания локальных изменений рельефа в результате эрозионных процессов. Полученные результаты были сопоставлены с данными спутников Sentinel-1/2 и показали высокую корреляцию с фактическими изменениями поверхности.

Интеллектуальное прогнозирование рельефа местности на основе данных БПЛА и ГИС ДЗЗ представляет собой перспективное направление пространственного анализа, особенно в условиях быстроменяющихся природных или техногенных процессов. Использование моделей глубокого обучения позволяет учитывать сложные взаимосвязи между пространственными характеристиками и создавать адаптивные предсказательные модели высокой точности. Разработка таких моделей актуальна для задач мониторинга,

планирования землепользования, предсказания наводнений и эрозии, а также для автономной навигации мобильных платформ [48].

Общая структура алгоритма

Цель: прогнозирование локальных изменений рельефа (напр. снижение высоты, эрозионные углубления) на основе временных рядов DEM и мультиспектральных индексов (NDVI, NDWI и др.).

Структура LSTM-алгоритма

1. Сбор и синхронизация данных:
 - DEM (цифровые модели рельефа) - по датам.
 - Индексы: NDVI, NDWI, GNDVI - из мультиспектральной съемки БПЛА или Sentinel.
 - Метки - значения высоты в следующем временном шаге.
2. Предобработка:
 - Интерполяция DEM до одинаковой сетки.
 - Нормализация NDVI/DEM.
 - Формирование временных окон (sliding windows).
3. Архитектура модели:
 - Вход: [timestep, ширина × высота × каналов]
 - LSTM слои
 - Полносвязный слой
 - Выход: прогноз высоты / изменения высоты
4. Обучение и оценка модели
5. Прогноз и визуализация

Пример кода на Python (TensorFlow / Keras, Приложение Д)

Полученный результат

Слой маски эрозии на основе порога изменений.

- Экспорт прогноза обратно в GeoTIFF (через rasterio).
- Добавление других индексов (NDWI, GNDVI).
- Построение карты изменений (Δ DEM).

Входные данные: Три DEM (с интервалом в 2 недели), соответствующие NDVI, RMSE модели на валидации: ~ 0.57 м

Визуально: эрозионные углубления, овраги и склоны предсказаны с погрешностью < 1 м.

Прогнозное DEM показывает динамику изменения склонов и просадок после ливней.

Структура LSTM-алгоритма для интеллектуального прогнозирования поведения сложных объектов

Сбор и синхронизация данных

Цель - формирование многовременного набора пространственных данных:

DEM (Digital Elevation Model) - цифровые модели рельефа, полученные по датам наблюдений (Sentinel-1 или дронов).

Индексы растительности:

NDVI - нормализованный индекс растительности, характеризует плотность зеленой массы;

NDWI - индекс влаги, отражает уровень увлажнения почв;

GNDVI - индекс фотосинтетической активности.

Метки (targets) - фактические значения высоты или показателей NDVI на следующий временной шаг ($t+1$), используемые для обучения.

Предобработка данных

Пространственная интерполяция: все DEM и индексы приводятся к одинаковому пространственному разрешению (10×10 м).

Нормализация: данные каждого канала масштабируются в диапазон $[0, 1]$ для ускорения обучения.

Временные окна (sliding windows): формируются последовательности данных длиной T , например, 10 шагов ($t-9 \dots t$), которые подаются на вход модели для прогнозирования шага $t+1$.

Архитектура модели (LSTM-сеть)

Входные параметры тензор размера $[\text{batch_size}, T, H, W, C]$, где

T - число временных шагов, H, W - размеры карты (высота и ширина),

C - количество каналов (DEM, NDVI, NDWI, GNDVI и др.).

Блок модели:

Conv2D слой - для выделения пространственных признаков (свертки).

ConvLSTM2D слои - для учета пространственно-временных зависимостей.

Flatten - Dense слой - для предсказания изменения высоты или NDVI.

Activation - линейная или ReLU-функция в зависимости от типа прогноза.

Выводы по вычислительной сложности

SIFT: высокая вычислительная сложность, медленно масштабируется; требует оптимизации и/или GPU.

ORB: низкая вычислительная сложность, легко масштабируется на сотни/тысячи снимков.

Hybrid: компромисс, точность лучше ORB, скорость ниже.

Масштабируемость: Линейная по числу изображений, если нет параллелизации. Хорошая при использовании multiprocessing или GPU.

Память зависит от числа дескрипторов на кадр и их размерности (SIFT 128 float, ORB 256 bits). На выходе получаем, карты прогнозируемых значений (изменение рельефа ΔDEM или рост растительности ΔNDVI).

Обучение и оценка модели

Функция потерь: MSE (Mean Squared Error) или MAE (Mean Absolute Error).

Метрики: алгоритм, R^2 , визуальная оценка карт прогнозов.

Регуляризация: dropout, batch normalization.

Оптимизатор: Adam (lr = 0.001).

Оценка: на тестовом наборе данных через временные срезы.

Прогноз.

Прогнозируются будущие карты DEM или NDVI (на 5–10 дней вперед).

Результаты отображаются на геоинформационной панели (Dash/Leaflet):

- анимированные карты изменений;
- временные графики трендов NDVI;
- зоны риска (эрозия, снижение растительности, оползни);
- экспорт прогнозов в GeoTIFF или shapefile

Пример кода ядра модели (фрагмент Python/Keras):

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import ConvLSTM2D, BatchNormalization,
Conv3D
model = Sequential([
    ConvLSTM2D(filters=64, kernel_size=(3, 3),
               activation='tanh', recurrent_dropout=0.2,
               input_shape=(10, 64, 64, 4), return_sequences=True),
    BatchNormalization(),
    ConvLSTM2D(filters=32, kernel_size=(3, 3),
               activation='tanh', recurrent_dropout=0.2,
               return_sequences=False),
    BatchNormalization(),
    Conv3D(filters=1, kernel_size=(3, 3, 3),
           activation='sigmoid', padding='same')
])
model.compile(optimizer='adam', loss='mse')
model.summary()
```

Созданная LSTM-структура формирует основу интеллектуальной информационной системы для анализа и прогнозирования динамики сложных природных объектов (рельефа, растительности или изменения поверхности) на базе данных ДЗЗ (рисунок 85).

Ортофотоплан

Карта

Атрибутивная информация

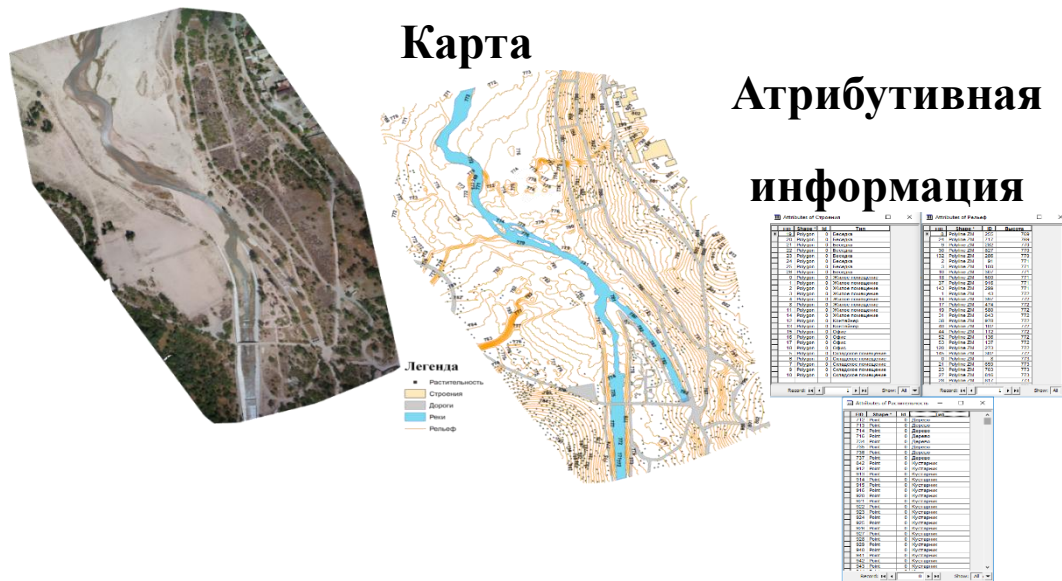


Рисунок 85 – Фрагменты геоинформационной системы

Система сочетает методы глубокого обучения, геоинформационного анализа и визуализации, обеспечивая автоматическую оценку и прогноз пространственно-временных изменений

Эффективность алгоритмов навигации и анализ шумов

Для оценки устойчивости алгоритма построения проективной матрицы к шумовым искажениям был проведен ряд испытаний. Результаты подтверждают, что:

- при нулевом уровне шума достигается высокая точность в определении углов и линейных смещений;
- при возрастании шума наблюдается пропорциональный рост ошибок, что требует дополнительных летных тестов для верификации [53].

Пример обработки реальных полетных данных с фильтрацией по Калману показан на рисунке 86. Детализация компонент движения продемонстрирована на рисунках 87–89.

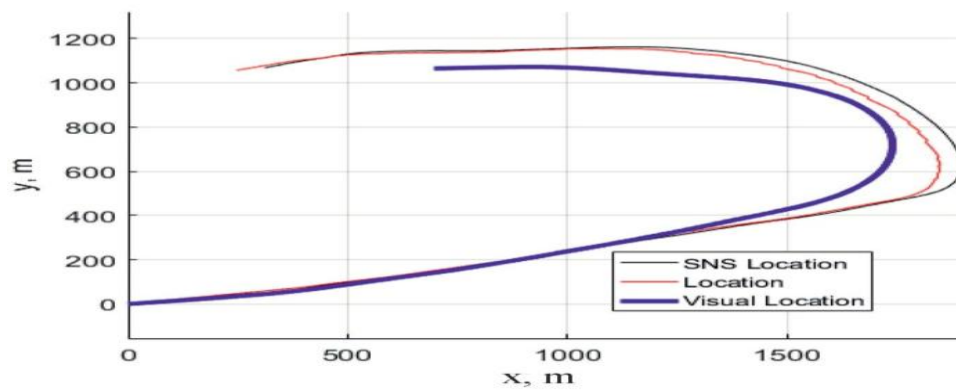


Рисунок 86 – Реальные полетные данные с оценкой положения БПЛА на основе фильтрации Калмана

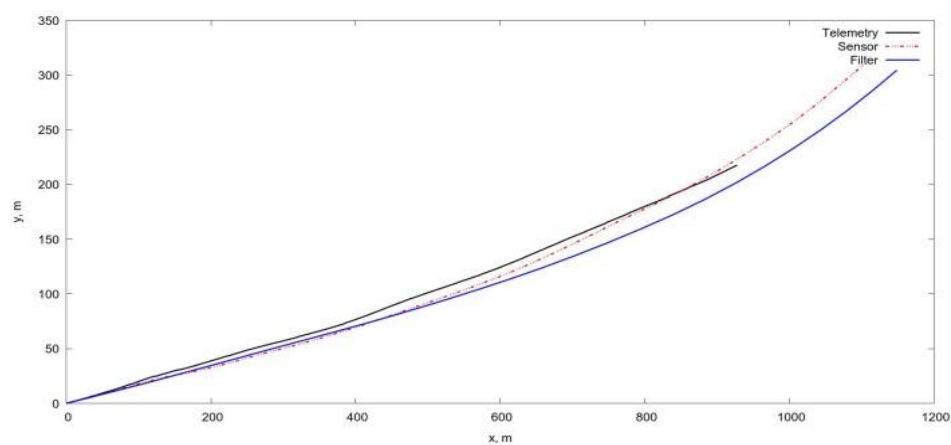


Рисунок 87 – Компоненты движения БПЛА

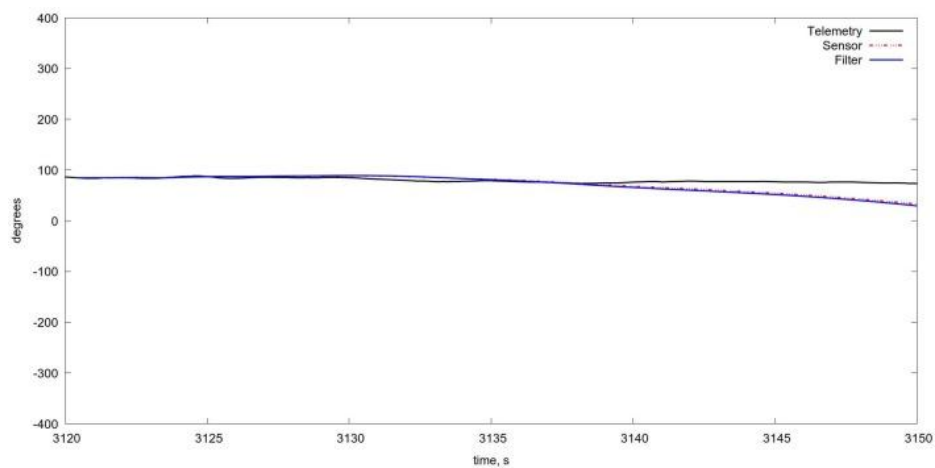


Рисунок 88 – Компоненты движения БПЛА

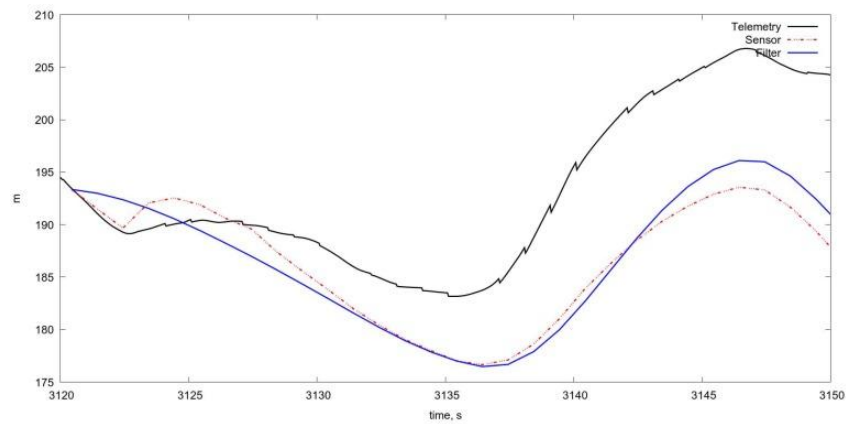


Рисунок 89 – Отслеживание угла рыскания

Статистическая оценка точности

В таблице 27 приведены оценки ошибок проективного преобразования по результатам моделирования.

Таблица 27 – Погрешности по алгоритму проективных матриц.

Количество кадров $\Delta t = 0,5 \text{ с}$	L_2 Норма реального сдвига от ИНС	L_2 Норма погрешности от ИНС и проективных матриц
1	17,9950	0,5238
2	16,6470	2,2434
3	17,4132	0,9915
4	16,4036	3,2009
5	17,2792	5,2980
6	16,2951	3,7480
7	17,7030	6,7479
8	16,4029	4,8240
9	17,8309	0,1152
10	16,1718	3,8627
11	17,2313	2,6757
12	15,8342	6,7456
13	17,2652	2,1774
14	15,7978	1,5271
15	17,4473	2,2599
16	15,9407	3,8367

Количество кадров $\Delta t = 0,5 \text{ с}$	L_2 Норма реального сдвига от ИНС	L_2 Норма погрешности от ИНС и проективных матриц
17	17,4862	5,3902
18	16,0020	1,9714
19	17,2286	2,1162
20	15,9406	3,4193

Рисунки 90–92 представляют оценку углов ориентации и координат в пространстве.

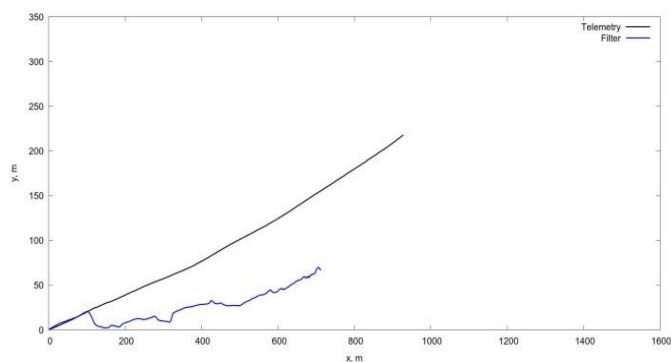


Рисунок 90 – Оценка углов ориентации

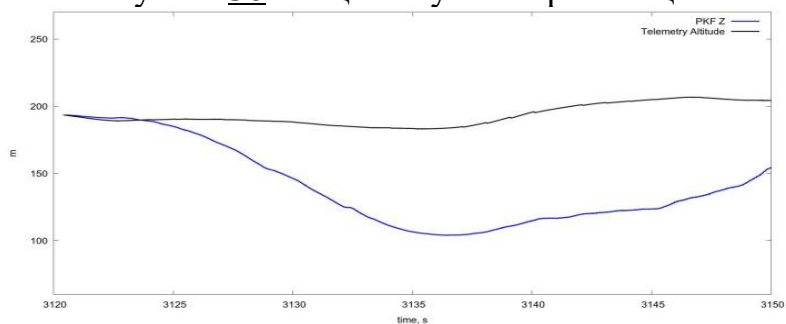


Рисунок 91 – Отслеживание координат X, Y

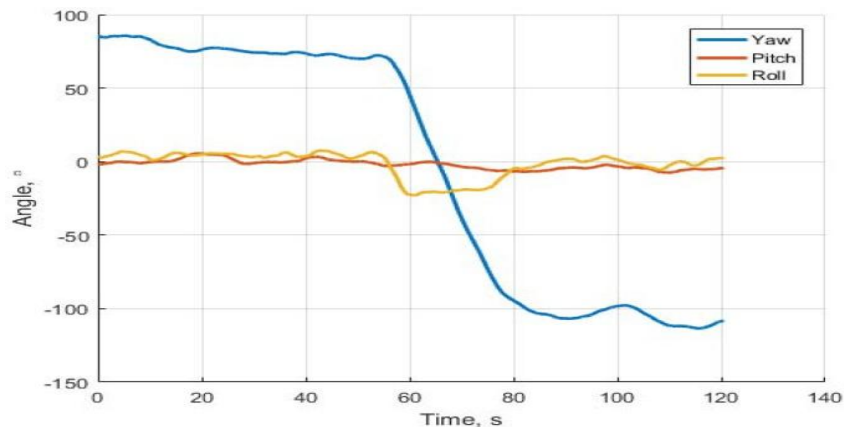


Рисунок 92 – Отслеживание Z-координаты

Угол рыскания в начальной фазе достигал ≈ 85 градусов (Таблица 28).

Таблица 28 - Основные параметры

Алгоритм	Теоретическая сложность на одно изображение	Особенности
SIFT	$O(N_{\text{pixels}} \times N_{\text{octaves}} \times N_{\text{levels}})$	Высокая вычислительная нагрузка, точность высокая, медленный
ORB	$O(N_{\text{keypoints}})$	Быстро, бинарные дескрипторы, меньше ресурсов, подходит для реального времени
Hybrid	Комбинация SIFT для ключевых точек и ORB для остальных	Баланс точности и скорости

Алгоритм, основанный на поэтапном расчете проективного преобразования между кадрами, потенциально может служить источником навигационной информации, дополняющим ИНС и ГНСС. Его успешность определяется точностью детектирования и сопоставления ключевых точек изображения. Однако накопление ошибок требует обязательной интеграции с другими системами, обеспечивающими фильтрацию и коррекцию.

- $N_{\text{images}} = 300$
- Размер изображения: 640×480 или больше
- Количество ключевых точек (ORB): 2000 на кадр
- Количество ключевых точек (SIFT): автоматически (~ 1000 – 5000 в зависимости от сцены)

Применение видеонавигации БПЛА на основе анализа изображений с поверхности Земли представляет собой многофакторную задачу, зависящую от рельефа, погодных условий и характеристик оборудования [49]. Наиболее перспективными направлениями в этом контексте являются:

- адаптивные алгоритмы выбора методов в зависимости от окружающей обстановки;
- слияние визуальных и инерциальных данных;
- использование фильтров Калмана и Байеса в реальном времени.

Интеграция видеоаналитики с традиционными навигационными средствами открывает путь к созданию устойчивых и автономных систем управления беспилотными платформами.

ЗАКЛЮЧЕНИЕ

Автор принимал непосредственное участие во всех этапах разработки - от постановки задачи, выбора архитектуры микроконтроллерных и программных решений, до проведения летных испытаний, обработки данных и построения ГИС-моделей. Диссертационная работа отражает оригинальные результаты, полученные лично автором в рамках заявленной темы.

Настоящая диссертационная работа посвящена разработке и исследованию интеллектуальной технологии управления БПЛА для решения задач дистанционного зондирования и фотограмметрической съемки, а также построению ГИС с высоким уровнем пространственной и тематической точности. Исследование базируется на синтезе современных методов автоматического управления, алгоритмов компьютерного зрения и концепций ИИ.

В рамках работы:

- Разработана интеллектуальная информационная система состоящая из 6 модулей
- Сформулирована и реализована архитектура интеллектуального аэросъемочного комплекса (АК), включающая микропроцессорные системы автопилотирования, навигации, управления затвором камеры и привязки снимков по данным инерциальной системы. Построенный прототип БПЛА гексакоптерного типа продемонстрировал высокую эффективность в задачах картографирования, мониторинга и построения 3D моделей.
- Разработаны и протестированы программно-аппаратные модули: модуль планирования маршрутов («Планировщик полетов»), микроконтроллерные блоки управления съемкой, модуль GPS-привязки и запись на SD-карту, а также программное обеспечение для извлечения телеметрических данных из IMU.
- Выполнена интеграция комплекса с системой фотограмметрической обработки, включая автоматизированное формирование ортофотопланов, цифровых моделей рельефа и 3D реконструкций. Созданы карты высот и тематические слои, включая растительность, гидрографию, объекты инфраструктуры.
- Разработана методология и реализована интеллектуальная технология повышения точности результатов аэросъемки. Применение оптимизированных параметров съемки (перекрытия, затвор, фокусировка, GPS/IMU-синхронизация) позволило добиться субдециметровой точности геопривязки.
- Предложены и исследованы алгоритмы визуальной навигации БПЛА с использованием методов сопоставления изображений. Проведена сравнительная оценка эффективности алгоритмов SIFT и ORB в задаче

определения навигационных параметров по эталонным изображениям, с применением фильтрации на основе RANSAC и фильтра Калмана. Установлено, что средняя ошибка позиционирования составляет около 3,18 м, а относительная ошибка - 19% на кадр, что указывает на необходимость интеграции с инерциальными и спутниковыми источниками данных.

– В качестве результата апробации технологии созданы и протестированы ГИС-системы по данным аэросъемки следующих территорий: искусственное водохранилище «Сайран», район «Алатау», карьеры и курганные комплексы. Эти системы включают ортофотопланы, цифровые модели рельефа, векторные слои и атрибутивные базы данных, реализованные в системах ArcGIS и CityEngine.

– Подтверждена применимость предложенной технологии для задач высокоточного мониторинга, в том числе при отсутствии сигнала спутниковой навигации. Выявлена высокая мобильность и экономическая эффективность предложенного подхода в сравнении с традиционными методами космической или авиационной съемки.

Ключевые научные результаты:

1. Разработан комплексный алгоритмический и аппаратный подход к управлению съемочным процессом БПЛА с интеллектуальной настройкой параметров.

2. Создана система автоматизированной привязки, обработки и визуализации аэросъемочных данных в формате ГИС.

3. Разработано программное обеспечение и микропроцессорные модули, реализующие интеллектуальные функции навигации, управления затвором, и синхронизации данных.

4. Проведено исследование алгоритмов видеоанализа (SIFT, ORB) в задачах навигации, подтверждена их применимость в условиях ограниченного спутникового сигнала.

5. Построены цифровые ортофотопланы, 3D модели и карты высот для ряда тестовых территорий, подтверждена высокая точность полученных данных.

6. Получены правовые охранные документы: 1 патент, 5 авторских свидетельств, 2 учебных пособия.

7. Подтверждена универсальность технологии, пригодной для применения в аэрокосмической, аграрной, транспортной и градостроительной сферах.

Результаты исследования нашли отражение в 36 научных публикациях, включая 12 статей в изданиях, рекомендованных КН МНВО РК, 3 в журналах из базы Scopus, 14 в материалах конференций и семинарах, а также в защищенных разработках, 1 патент и 6 авторских свидетельств.

Диссертационной работе заложена научно-методологическая и прикладная основа для построения интеллектуальных систем управления аэросъемкой с

БПЛА, и она может служить базой для последующих исследований в области гибридных навигационных решений и интеграции видеонаблюдения с системами управления и анализа сложных объектов.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Республика Казахстан. Президент (2024. К.К.Токаев). Послание Главы государства Касым-Жомарта Токаева народу Казахстана от 2 сентября 2024 года. <https://www.akorda.kz/ru/poslanie-glavy-gosudarstva-kasym-zhomarta-tokaeva-narodu-kazahstana-spravedlivyy-kazahstan-zakon-i-poryadok-ekonomicheskii-rost-obshchestvenn> .
2. Y. Fu et al., Remote Sensing Time Series Analysis: A Review of Data and Applications, Remote Sensing (review). [spj.science.org](https://www.spj.science.org)
3. Z. Chen, Long sequence time-series forecasting with deep learning (survey), 2023. ScienceDirect
4. F. Samadzadegan et al., A critical review on multi-sensor and multi-platform remote sensing data fusion, Int. J. Remote Sensing, 2024–2025. [tandfonline.com](https://www.tandfonline.com)
5. P. Wu et al., UAV target tracking: a survey, Artificial Intelligence Review, 2025. SpringerLink
6. S. Khan et al., On GPS spoofing of aerial platforms: a review of threats and countermeasures, 2021 (PMC).
7. R. Liu et al., Using UAV Data to Improve Satellite Monitoring: upscaling approaches, Land, MDPI, 2024. MDPI
8. A. Allu et al., Fusion of Satellite and UAV Imagery for Crop Monitoring, ISPRS Annals (2025). [isprs-annals.copernicus.org](https://www.isprs-annals.copernicus.org)
9. Y. Cao et al., Research on Trajectory Prediction Algorithm Based on Intention Recognition, Drones (MDPI) / related 2025 papers. MDPI
10. J. Zhang et al., Novel UAV trajectory prediction methods (QCNet etc.), Aerospace/Elsevier 2025. ScienceDirect
11. SE Meheret et al., Systematic review on spoofing and jamming in UAV navigation, JATM / 2025. [jatm.com.br](https://www.jatm.com.br)
12. Ainakulov Zh.Zh., Makarenko N.G., Paltashev T.T. «Intelligent Monitoring on the Basis of a UAV for Assessing and Predicting the Condition of Objects with Complex Relief». International Journal of Applied Mathematics and Informatics, 13, 2019, pp. 54-58. [naun.org+1](https://www.naun.org+1)
13. Razakova M.G., Kuzmin A.G., Ainakulov Zh.Zh. и др. «Development of Hardware and Software Architecture for Analysis and Processing of the Field Data». International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences – ISPRS Archives, 43(B2), 2020, pp. 1253-1258. [jatit.org](https://www.jatit.org)
14. Ataeva, O. M., Serebryakov, V. A., & Tuchkova, N. P. (2018). Approaches to the organization of mathematical knowledge in the construction of subject thesauri of various branches of mathematics. Paper presented at the CEUR Workshop Proceedings, 2260 42-54.
15. Ainakulov Zh.Zh., Makarenko N.G., Paltashev T.T. Experience of modeling mining landscapes by using methods of intelligent monitoring, The journal «Current problems

- in remote sensing of the Earth from space», 2018. – Volume – №7. – P. 43-50. DOI: 10.21046/2070-7401-2018-15-7-43-50
16. Aleksieva-Petrova, A., Mladenova, I., Dimitrova, K., Iliev, K., Georgiev, A., & Dyankova, A. (2022). Earth-observation-based services for national reporting of the sustainable development goal Indicators-Three showcases in bulgaria. *Remote Sensing*, 14(11)
 17. Hasanin, T. (2023). A spatio-demographic analysis over twitter data using artificial neural networks doi:10.1007/978-981-19-4052-1_55
 18. Liu, J., Wang, D., Lin, Q., & Deng, M. (2023). Risk assessment based on FMEA combining DEA and cloud model: A case application in robot-assisted rehabilitation. *Expert Systems with Applications*, 214 doi:10.1016/j.eswa.2022.119119
 19. Mosayebian, M. E. (2023). A new approach for modeling wind power in reliability studies. *Journal of Operation and Automation in Power Engineering*, 11(2), 144-150. doi:10.22098/JOAPE.2023.10081.1713 12.
 20. Костенко И.П. Введение в вероятностное прогнозирование. – М.: Институт компьютерных исследований, 2004. – 316 с
 21. Campora, J. P., Chen, S., Erwig, M., & Walkingshaw, E. (2022). Migrating gradual types. *Journal of Functional Programming*, 32(1) doi:10.1017/S0956796822000089
 22. Ainakulov Zh.Zh., Makarenko N.G., Paltashev T.T. Experience of modeling mining landscapes by using methods of intelligent monitoring. The journal «Current problems in remote sensing of the Earth from space», 2018. –Volume 15. –№7. – P. 43-50. (SCOPUS) (Accepted: 04.09.2018, DOI: 10.21046/2070-7401-2018-15-7-43-50) (SJR = 0.260)
 23. Разакова М.Г., Кузьмин А.Г., Федоров И.О., Ергалиев Р.К. Параметры нижнего приземного слоя воздуха с использованием беспилотного летательного аппарата. Сборник тезисов докладов шестнадцатой Всероссийской открытой конференции "Современные проблемы дистанционного зондирования Земли из космоса", Москва, ИКИ РАН, РФ, 12-16 ноября 2018 г. (http://smiswww.iki.rssi.ru/d33_conf), – С. 431, (DOI 10.21046/2070-16DZZconf-2018a)
 24. Lacagnina, C., Doblas-Reyes, F., Larnicol, G., Buontempo, C., Obregón, A., Costa-Surós, M., . . . Díez, M. G. (2022). Quality management framework for climate datasets. *Data Science Journal*, 21(1) doi:10.5334/dsj-2022-010
 25. Christofi, D.; Mettas, C.; Evagorou, E.; Stylianou, N.; Eliades, M.; Theocharidis, C.; Chatzipavlis, A.; Hasiotis, T.; Hadjimitsis, D. A Review of Open Remote Sensing Data with GIS, AI, and UAV Support for Shoreline Detection and Coastal Erosion Monitoring. *Applied Sciences*, 2025, 15 (9), 4771. DOI: 10.3390/app15094771. [MDPI+1](#)
 26. Lee, J.; Sanjmyatav, B.; Kang, I. U.; Go, A.; Diimaajav, E.; Baatar, G.; Enkhbayar, M.; Han, J. UAV-Based Remote Sensing, GIS, and Its Application in Archaeology: A Case Study of the Xiongnu Elite Burial Complex at Gol Mod-II, Mongolia. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-

- M-2, 2025, pp. 201-208. DOI: 10.5194/isprs-annals-X-M-2-2025-201-2025. isprs-annals.copernicus.org
27. Colpaert, A. Satellite and UAV Platforms, Remote Sensing for Geographic Information Systems. Sensors, 2022, 22 (12), 4564. DOI: 10.3390/s22124564. [MDPI](https://www.mdpi.com/1424-6460/22/12/4564)
 28. Luo, W.; Zhang, G. (Eds.) Editorial: Advances and applications of geospatial modeling and analysis in digital twins. Frontiers in Earth Science, 2023, 11, 1226466. DOI: 10.3389/feart.2023.1226466. [Frontiers](https://www.frontiersin.org/articles/10.3389/feart.2023.1226466)
 29. Wang, Y.; Yang, Y.; Wang, S.; Yang, Y.; Zhang, R. A review on the architecture construction of remote sensing data from unmanned aerial vehicle networking. Progress in Geography, 2021, Vol. 40, Issue 9, pp. 1467-1479. DOI: 10.18306/dlkxjz.2021.09.003. [progressingearth.com](https://www.progressingearth.com)
 30. Remote Sensing | Special Issue: Advances in Remote Sensing and Digital Twins for Management of Civil Infrastructure Assets. Remote Sensing (MDPI), ISSN 2072-4292. (Special Issue Call). [MDPI](https://www.mdpi.com/2072-4292/special_issues/rsdtw)
 31. Dong, W., Chen, Z., Chen, J., Ting, Z. J., Zhang, R., Ji, G., & Zhao, M. (2022). A novel method for the estimation of higher heating value of municipal solid wastes. Energies, 15(7) doi:10.3390/en15072593
 32. [Farhad Samadzadegan](https://orcid.org/0000-0003-3439-5648) a School of Surveying and Geospatial Engineering, University College of Engineering, University of Tehran, Tehran, Ira <https://orcid.org/0000-0003-3439-5648> [Ahmad Toosi](https://www.researchgate.net/profile/Ahmad-Toosi), [Farzaneh Dadrass Javan](https://www.researchgate.net/profile/Farzaneh-Dadrass-Javan)
Pages 1327-1402 | Received 11 Aug 2024, Accepted 03 Nov 2024, Published online: 11 Dec 2024. [Cite this article. https://doi.org/10.1080/01431161.2024.2429784](https://doi.org/10.1080/01431161.2024.2429784)
 33. Xia, K., Shin, M., Chung, W., Kim, M., Lee, S., & Son, H. (2022). Landing a quadrotor UAV on a moving platform with sway motion using robust control. Control Engineering Practice, 128 doi:10.1016/j.conengprac.2022.105288
 34. Айнакулов Ж.Ж., и др. «Developing a comprehensive solution for collecting and managing data from machines by different manufacturers». AIP Conference Proceedings, 3033(1), 2024, Article 020013. [northkazdss.kaznaru.edu.kz](https://www.northkazdss.kaznaru.edu.kz)
 35. Айнакулов Ж.Ж., Кузьмин А.Г., Мухамедгалиев А.Ф., Разакова М.Г., Смирнов В.В. Управление временем срабатывания затвора фотокамеры беспилотного летательного аппарата Проблемы информатики. 2016 г. – №1. – С. 84-92
 36. Макаренко Н.Г., Смирнов В.В., Кузьмин А.Г., Разакова М.Г., Кожаев Ж.Т. Аэрофотосъемка открытых горнорудных месторождений с использованием БПЛА. Вестник КазНУ. – Алматы, 2017. – №4 (122). – С. 371-376.
 37. ZHARAS AINAKULOV, KAYRAT KOSHEKOV, NATALYA ASTAPENKO, ILDAR PIRMANOV, ABAY KOSHEKOV. THE EXPERIENCE OF INTRODUCING DIGITAL TWINS INTO THE EDUCATIONAL PROCESS ON THE EXAMPLE OF TRAINING IN THE REPAIR OF AIRCRAFT EQUIPMENT UNITS. Journal of Theoretical and Applied Information Technology. 30th June 2023 - Vol. 101. No. 12 – 2023

38. Li, J., Zhang, Q., Deng, W., Tang, Y., Zhang, X., & Wu, Q. (2022). Source direction finding and direct localization exploiting UAV array with unknown gain/phase errors. *IEEE Internet of Things Journal*, 9(21), 21561-21569. doi:10.1109/JIOT.2022.3181450
39. Zhang, L., Li, K., Qi, Y., & Wang, F. (2021). Local feature extracted by the improved bag of features method for person re-identification. *Neurocomputing*, 458, 690-700. doi:10.1016/j.neucom.2019.12.142
40. **Christofi, D. et al. (2025).** *A Review of Open Remote Sensing Data with GIS, AI, and UAV Support for Shoreline Detection and Coastal Erosion Monitoring.* *Applied Sciences*. [MDPI](#)
41. **Pal, O.K. et al. (2024).** *In-depth review of AI-enabled unmanned aerial vehicles: trends, vision, and challenges.* *Discover Artificial Intelligence*. DOI: 10.1007/s44163-024-00209-1. [SpringerLink](#)
42. **Tang, G. et al. (2023).** *A Survey of Object Detection for UAVs Based on Deep Learning.* *Remote Sensing*. [MDPI](#)
43. **Quamar, M.M. et al. (2023).** *Advancements and Applications of Drone-Integrated GIS.* *Remote Sensing*. [MDPI](#)
44. **Kazanskiy, N. et al. (2025).** *A Comprehensive Review of Remote Sensing and Artificial Intelligence.* *Sensors*. DOI: 10.3390/s25195965. [MDPI+1](#)
45. **Mai, G. et al. (2025).** *Towards the next generation of Geospatial Artificial Intelligence.* *Computers, Environment and Urban Systems* (review). [ScienceDirect](#)
46. **Zhao, C. et al. (2024).** *Deep learning-based object detection in maritime UAV imagery.* *Pattern Recognition Letters* / related review. [ScienceDirect+1](#)
47. **Anam, I. et al. (2024).** *A systematic review of UAV and AI integration for targeted crop management.* *Computers and Electronics in Agriculture*. [ScienceDirect](#)
48. **Al-Darraj, I. et al. (2023/2025).** *Integrating Drones With Digital Twins for Aerial Remote Sensing.* *Radio Science* / *Remote Sensing* (AGU/Wiley). agupubs.onlinelibrary.wiley.com
49. Gao, J., & Sun, Z. (2022). An improved ASIFT image feature matching algorithm based on POS information. *Sensors*, 22(20) doi:10.3390/s22207749
50. Цинь Т., Ли П., Шен С. Перемещение, глобальная оптимизация и объединение карт для монокулярного визуально-инерционного SLAM. В: Материалы международной конференции IEEE по робототехнике и автоматизации. Брисбен: HKUST; 2018.
51. Burdziakowski P. Low Cost Hexacopter Autonomous Platform for Testing and Developing Photogrammetry Technologies and Intelligent Navigation Systems; Proceedings of the 10th International Conference on “Environmental Engineering”; Vilnius, Lithuania. 27–28 April 2017; pp. 1–6. [Google Scholar]

52. Miller B., Miller G., Semenikhin K. Optimization of the Data Transmission Flow from Moving Object to Nonhomogeneous Network of Base Stations. IFAC PapersOnLine. 2017;50:6160–6165. doi: 10.1016/j.ifacol.2017.08.981. [CrossRef] [Google Scholar]
53. Гаганов В. Инвариантные алгоритмы сопоставления точечных особенностей на изображения // Компьютерная графика и мультимедиа. № 7 (1); URL: http://cgm.computer-graphics.ru/issues/issue17/invariant_features (дата обращения: 03.11.2022).
54. Ovchinkin A.A., Ershov E.I. The algorithm of epipole position estimation under pure camera translation. Sens. Syst. 2018;32:42–49. (In Russian) [Google Scholar]
55. Hong, Chenghua & Ge, Han & Fang, Cunzhong & Zhao, Xuyang & Liu, Ning & Wang, Bin & Zhao, Yang. (2022). Rapid Assembly Modeling in Computer Aided Design System. Journal of Physics: Conference Series. 2146. 012022. 10.1088/1742-6596/2146/1/012022.
56. <http://www.iah.bbsrc.ac.uk/phd/gisruk95.html>
57. <http://www.brgm.fr/geofrance3d/geofrance3d.html>
58. http://www.ncgia.ucsb.edu/conf/SANTA_FE_CD-ROM/sf_papers/hutchinson_michael_dem/local.html
59. http://www.geog.le.ac.uk/jwo/research/dem_char/thesis. Wikipedia. Высотомер. URL <http://ru.wikipedia.org/wiki/Высотомер>
60. OpenCV – open source computer vision library, URL: <http://opencv.org>
61. RoboCraft. OpenCV шаг за шагом. URL: <http://robocraft.ru/page/opencv>
62. ArduPilot Mega (APM) 2.6 на квадрокоптер [Электронный ресурс]: Полетный контроллер ArduPilot Mega (APM) 2.6. URL: <http://ardupilot-mega.ru>. (дата обращения 01.05.2015) . KAZUS.RU [Электронный ресурс]:
63. Управление затвором фотоаппарата через COM порт компьютера (вознаграждение) URL: <http://kazus.ru/forums/showthread.php?t=101370> (дата обращения 01.05.2015)
64. Аэрофотосъемка и ортофотоплан с БПЛА беспилотного аппарата от производителя [Электронный ресурс]: Аэрофотосъемка с БПЛА - ортофотоплан URL: <http://unmanned.ru/service/aerophoto.htm>
65. Ainakulov, Z., Kurmankulova, G., Schüle, H., Ainakulova, Z. Algorithms and tools for analysing and planning experiments in the agro-industrial complex. AIP Conference Proceedings Эта ссылка отключена., 2024, 3033(1), 020009. <https://www.scopus.com/authid/detail.uri?authorId=57606352400>. <https://doi.org/10.1063/5.0188480>
66. Ainakulov, Z., Kurmankulova, G., Schüle, H., Ainakulova, Z. Algorithms and tools for analysing and planning experiments in the agro-industrial complex. AIP Conference Proceedings Эта ссылка отключена., 2024, 3033(1), 020009. <https://www.scopus.com/authid/detail.uri?authorId=57606352400>. <https://doi.org/10.1063/5.0188480>

68. Кузьмин А.Г. Геоинформационное обеспечение агротехнологии точного земледелия на основе беспилотных летательных аппаратов. Материалы международной научной конференции «Информатика и прикладная математика», посвященная 25-летию независимости Республики Казахстан и 25-летию Института информационных и вычислительных технологий», – 1 том, – Алматы, 21-24 сентября 2016 г., – С. 47-53
69. Ainakulov, Z., и др. «Experience of modeling mining landscapes by using methods of intelligent monitoring». Современные проблемы дистанционного зондирования Земли из Космоса, 15(7), 2018, pp. 43-50. official.satbayev.university.kz
70. Ainakulov, Z., и др. «Developing a comprehensive solution for collecting and managing data from machines by different manufacturers». AIP Conference Proceedings, 3033(1), 2024, Article 020013. northkazdss.kaznaru.edu.kz
71. **Valencia, E. et al. (2025).** *An Open-source UAV Digital Twin framework: A Case Study.* Control Engineering Practice (или Robotics journal). [SpringerLink](https://www.springerlink.com)
72. **Aierken, N. et al. (2024).** *A review of unmanned aerial vehicle based remote sensing for crop monitoring.* Agricultural Systems / Field Crops. [ScienceDirect](https://www.sciencedirect.com)
73. **Fauzi, C. et al. (2024).** *A Review: Geospatial Artificial Intelligence (Geo-AI).* Конференц. материалы / proceedings. [Атлантис Пресс](https://www.atlantis-prices.com)
74. **Pal, P. et al. (2025).** *Unmanned aerial system and machine learning driven Digital Twin for crop monitoring.* Computers and Electronics in Agriculture. [ScienceDirect](https://www.sciencedirect.com)
75. **Collection of practical case studies (2024).** *UAV + GIS applied case studies (forest, agriculture, infrastructure).* (спецвыпуски Remote Sensing / ISPRS). [MDPI+1](https://www.mdpi.com)
76. **Special Issues: Remote Sensing - Digital Twins & Infrastructure (2025).** [MDPI+1](https://www.mdpi.com)
77. **Survey / Review on data fusion (multimodal RS + UAV) (2024).** [MDPI+1](https://www.mdpi.com)
78. **Generative AI for Urban Digital Twins (Xu et al., 2024–2025, препринты и журнальные статьи).** [sciprofiles.com+1](https://sciprofiles.com)
79. **Methods for real-time inference on UAVs (edge AI).** Pal 2024 и Tang 2023). [SpringerLink+1](https://www.springerlink.com)
80. **Validation and benchmarking studies (2024).** [MDPI+1](https://www.mdpi.com)
81. Startbase - Система активизации и повышения результативности инновационного процесса - Продукция / Обзор продукта / Картографический аэрофотосъемочный комплекс на базе БПЛА [Электронный ресурс]: Картографический аэрофотосъемочный комплекс на базе БПЛА URL: <http://www.startbase.ru/products/36455/>
82. ГЕОСалют [Электронный ресурс]: БПЛА на службе геодезистов URL: <http://geosalut.pulscen.ru/news/246836>

*В процессе подготовки данной рукописи автор(ы) ограниченно использовали системы искусственного интеллекта исключительно для вспомогательных задач

(проверка правописания и незначительное редактирование текста). Основное содержание, научные результаты и выводы сформированы автором(ами) самостоятельно. После использования инструмента текст был проверен и отредактирован. Автор(ы) несут полную ответственность за содержание публикации.

Публикации

Список опубликованных научных работ
соискателя степени доктора философии (PhD) Айнакулова Ж.Ж.
Специальность: 6D070300 - «Информационные системы»
НАО «Казахский Агротехнический исследовательский университет им.С.Сейфуллина»

№	Наименование трудов	Характер трудов	Выходные данные	Объем п.л	Авторы
Статьи в журналах, включаемых в список, рекомендованный Комитетом по контролю в сфере образования и науки Министерства науки и высшего образования РК					
1	Исследование 3D форматов хранения данных в прикладных интеллектуальных системах	Печ	Вестник КазНПУ (Серия физико-математические науки). – Алматы, 2016 г., – №3(55). – С. 121-127.	0,3	Айнакулов Ж.Ж., Курманкулова Г.Е.
2	Аэрофотосъемка открытых горнорудных месторождений с использованием БПЛА	Печ	Вестник КазНПУ. – Алматы, 2017. – №4 (122). – С. 371-376.	0,3	Айнакулов Ж.Ж., Макаренко Н.Г., Смирнов В.В., Кузьмин А.Г., Разакова М.Г., Кожаев Ж.Т.
3	Анализ и прогноз результатов применения интеллектуальных обучающих систем	Печ	Вестник КазГАСА. – Алматы, 2018. – №3 (69). – С. 211-217.	0,3	Айнакулов Ж.Ж., Курманкулова Г.Е., Дильмагамбетова Б.М.
4	Разработка модифицированной системы управления БПЛА для оперативного мониторинга состояния природных и техногенных территориальных комплексов	Печ	«Вестник КазНПУ», – Алматы, 2020. – №6. – С 27-33.	0,3	Айнакулов Ж.Ж., Разакова М.Г., Кузьмин А.Г.
5	Виртуальное моделирование и мониторинг деталей авиационной техники	Печ	ВЕСТНИК КазНПУ им. Абая, серия «Физико-математические науки», №3(75), 2021 г., - С. 35-43. https://doi.org/10.51889/2021-3.1728-7901.04 .	0,5	Ж.Ж. Айнакулов, Д.В. Зуев, Б.К. Сейдахметов, И.О. Федоров, К.Т. Кошеков

Соискатель

Список подтверждаю

Ученый секретарь



Ж.Ж. Айнакулов

Г.М. Дерипсалдина

6	Совокупность отличительных свойств этапов разработки программных средств VR	Печ	Вестник Satbayev University, 143(5), 157–164. https://doi.org/10.51301/vest.su.2021.i5.20	0,4	Пирманов, И., Кошеков, К., Федоров, И., Сейдахметов, Б., Зуев, Д., & Айнакулов, Ж.
7	Методика разработки обучающих программных средств VR для ремонта авиационной техники	Печ	Вестник КазНПУ имени Абая. Серия: Физико-математические науки. 78, 2 (июн. 2022), 143–149. DOI: https://doi.org/10.51889/2022-2.1728-7901.18 .	0,3	Пирманов И., Айнакулов Ж., Зуев Д., Астапенко Н., Федоров И. и Кошеков К.
8	Определение неопределенности в летном движении беспилотного летательного аппарата с помощью регуляторного синтеза.	Печ	Вестник КазАТК, 122(3), 76–82. https://doi.org/10.52167/1609-1817-2022-122-3-76-82	0,3	Керибаева, Т., Айнакулов, Ж., Нурланулы, А., & Дауренбаева, Н.
9	Алгоритм проектирования VR-приложений для технологических процессов ремонта авиационной техники	Печ	«ВЕСТИК ВКТУ» № 3, 2022, 150-167. DOI 10.51885/1561-4212_2022_3_150	0,8	К.Т. Кошеков, Ж.Ж. Айнакулов, И.А. Пирманов, Н.В. Астапенко, К. Алибеккызы*, А.Т. Байдильдина
10	Методы и инструменты оперативной психодиагностики и для выявления и предотвращения актов незаконного вмешательства в деятельность гражданской авиации	Печ	Вестник КазНПУ имени Абая. Серия: Психология, 72(3). https://doi.org/10.51889/4558.2022.94.97.025	0,5	Ainakulov, Z., Bogacheva, M., Suleimanova, E., Beisenbayeva, A., Koshekov, K., & Kumankulova, G.

Соискатель

Список подтверждено

Ученый секретарь



Ж.Ж. Айнакулов

Г.М. Дерипсалдина

11	Анализ специфики работы инспекторов-профайлеров в гражданской авиации	Печ	Вестник КазНПУ имени Абая. Серия: Психология, 73(4). https://doi.org/10.51889/9612.2022.74.43.002	0,2	Айнакулов, Ж., Богачева, М., Сулейманова, Е., Бейсенбаева, А., Кошекков, К., & Курманкулова, Г.
12	Роль валоканав в сельском хозяйстве: мелиорация, борьба с эрозией и сохранение влаги	Печ	Вестник науки Казахского агротехнического исследовательского университета имени С. Сейфуллина: междисциплинарный, (1[124]), 13–26. https://doi.org/10.51452/kazatu.2025.1(124).1807	0,7	Айкешев, Б. М., Айнакулов, Ж. Ж., & Нурушев, М. Ж.
Статьи в международных рецензируемых научных журналах с ненулевым импакт-фактором					
1	Управление временем срабатывания затвора фотокамеры беспилотного летательного аппарата	Печ	Проблемы информатики. 2016 г. – №1. – С. 84-92.	0,5	Айнакулов Ж.Ж., Кузьмин А.Г., Мухамедгалиев А.Ф., Разакова М.Г., Смирнов В.В.
2	Intelligent Monitoring on the Basis of a UAV for Assessing and Predicting the Condition of Objects with Complex Relief	Печ	International Journal of Applied Mathematics and Informatics, Volume 13, 2019. P. 54-58 (zbMath) (ISSN: 2074-1278).	0,2	Ainakulov Zh.Zh., Makarenko N.G., Paltashev T.T.
3	Development and implementation of visual data models from UAV	Печ	ISJ Theoretical & Applied Science, 04 (96), 429-438. Soi: http://s-o-i.org/1.1/TAS-04-96-86 Doi: https://dx.doi.org/10.15863/TAS.2021.06.96.86	0,5	Ainakulov Zh.Zh., Kurmankulova G.E., Zhumakozhaeva A.G.
4	Hardware-Software Complex of Interactive Training Programs on Aircraft Repair Processes Based on 3D and VR	Печ	Journal of Theoretical and Applied Information Technology 28th February 2022 – Vol. 100. No. 04 – 2022. P. 1148-1157.	0,5	Ildar Pirmanov, Zharas Ainakulov, Natalya Astapenko, Kairat Koshekov, Igor Fedorov

Соискатель

Список подтверждаю

Ученый секретарь



Ж.Ж. Айнакулов

Г.М. Дерипсалдина

5	Experience of Connecting Sensors to the Controller Based on the Arduino Board for Use on Multicopters	Печ	Journal of Theoretical and Applied Information Technology 31st March 2022 – Vol. 100, No. 06 – 2022. P. 1827-1835.	0,5	Talshyn Keribayeva, Zharas Ainakulov , Rustam Yergaliyev, Gulzhan Kurmankulova, Igor Fedorov, Raziyam Anayatova
6	Modeling the process of implementation of attacks and estimation of its probabilistic parameters	Печ	ISJ Theoretical & Applied Science, 02 (118), 539-544. Soi: http://s-o-i.org/1.1/TAS-02-118-47 Doi: https://dx.doi.org/10.15863/TAS.2023.02.118.47	0,3	Zhaksylykova, A., Ainakulov, Zh. , & Kurmankulova, G.
7	The experience of introducing digital twins into the educational process on the example of training in the repair of aircraft equipment units	Печ	Journal of Theoretical and Applied Information Technology. 30th June 2023 - Vol. 101, No. 12 – 2023.	0,6	Zharas Ainakulov , Kayrat Koshekov, Natalya Astapenko, Ildar Pirmanov, Abay Koshekov
8	Machine learning methods in planning experiments and data analysis in the agricultural industry	Печ	ISJ Theoretical & Applied Science, 04 (144), 38-44	0,3	Ainakulov, Zh. , Iskakova, A., & Kurmankulova, G.
9	Development of a Hybrid Quadrotor Control Framework Integrating Classical Control and Vision-Based Deep Learning for Autonomous Takeoff, Landing, and Object Tracking	Печ	Journal of Cultural Analysis and Social Change, 10(4), 1284–1307. https://doi.org/10.64753/jcasc.v10i4.3013	1,1	Nguyen, D., Kinzhikeyev, S., Duong, H., Doan, M., Yermoldina, G., Ainakulov, Z. , Suimenbayeva Z., Bebenin, A.

Сонскатель

Список подтверждаю

Ученый секретарь



Ж.Ж. Айнакулов

Г.М. Дерипсалдина

Статьи в международных рецензируемых научных журналах, имеющие в базе данных Scopus (Скопус) показатель проценты по Cite Score (Сайт Скоп) не менее 25					
1	Experience of modeling mining landscapes by using methods of intelligent monitoring.	Печ	The journal «Current problems in remote sensing of the Earth from space», 2018. – Volume 15. – №7. – P. 43-50. (процентиль 36) (Accepted: 04.09.2018, DOI: 10.21046/2070-7401-2018-15-7-43-50) (SJR = 0.260).	0,4	Ainakulov Zh.Zh., Makarenko N.G., Paltashev T.T.
2	Development of an advanced ai-based model for human psychomotional state analysis.	Печ	Eastern-European Journal of Enterprise Technologies, 6 (4 (126)), 39–49. doi: https://doi.org/10.15587/1729-4061.2023.293011 . (процентиль 54)	0,5	Ainakulov, Z., Koshekov, K., Savostin, A., Anayatova, R., Seidakhmetov, B., Kurmankulova, G.
3	Analysis of World Experience and Experimental Implementation of Unmanned Radio Intelligence Systems.	Печ	Transport and Telecommunication, 27(1), 73-92. https://doi.org/10.2478/ttj-2026-0007 . (процентиль 62)	1	Gulnaz, Y., Zharas, A., Zhanna, S., & Andrey, B.
Международные конференции, статьи в сборниках					
1	Methods of calculating landslide volume using remote sensing data	Печ.	E3S Web Conf. Volume 149, 2020 Regional Problems of Earth Remote Sensing (RPERS 2019) (SCOPUS) (DOI https://doi.org/10.1051/e3sconf/202014902009)	0,2	Zharas Ainakulov, Maira Razakova, Alexandr Kuzmin, Igor Fedorov, Rustam Yergaliev
2	Development of hardware and software architecture for analysis and processing of the field data	Печ.	Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLIII-B2-2020, 1253–1258 (SCOPUS, WOS), https://doi.org/10.5194/isprs-archives-XLIII-B2-2020-1253-2020 , 2020.	0,3	Ainakulov Zh.Zh., Razakova, M.G., Kuzmin, A.G., Fedorov, I.O., Yergaliev, R.K.
3	Algorithms and tools for analysing and planning experiments in the agro-industrial complex	Печ.	AIP Conf. Proc. 12 January 2024; 3033 (1): 020009. https://doi.org/10.1063/5.0188480	0,6	Zharas Ainakulov, Gulzhan Kurmankulova, Heinrich Schüle, Zhadra Ainakulova

Соискатель

Список подтверждаю

Ученый секретари

Ж.Ж. Айнакулов

Г.М. Дерипсалдина

4	Developing a comprehensive solution for collecting and managing data from machines by different manufacturers	Печ.	AIP Conf. Proc. 12 January 2024; 3033 (1): 020013. https://doi.org/10.1063/5.0188481	0,5	Zharas Ainakulov, Gulzhan Kurmankulova, Serik Ospanov, Zhadra Ainakulova, Heinrich Schüle, Kulmuhanbet Akhmetov
5	Economic and mathematical modelling of estimating the use of basic production resources of agricultural formations	Печ.	AIP Conf. Proc. 12 January 2024; 3033 (1): 020022. https://doi.org/10.1063/5.0188482	0,8	Zharas Ainakulov, Kulmuhanbet Akhmetov, Serik Ospanov, Gulzhan Kurmankulova, Aizhan Tengaeva, Heinrich Schüle, Togzhan Kurmanbek
Интеллектуальная собственность в государственной регистрации, патенты и предварительные патенты, другие статьи					
1	Shutter release mechanism of a digital camera mounted on a UAV for aerial photography	Пат	Patent № 4609 For Utility Model 2019/0653.2 Date of the registration in the State Register of Utility Models of the Republic of Kazakhstan: 15.01.2020.	1	Ainakulov Zh.Zh., Razakova M.G., Kuzmin A.G., Smimov V.V.
2	Программа для ЭВМ «CAMERA»	Автор. свид	Свидетельство о внесении сведений в государственный реестр прав на объекты, охраняемые авторским правом № 16867 от «22» апреля 2021 года.	1	Ж.Ж. Айнакулов, Г.Е. Курманкулова.
3	Программа для ЭВМ «Планировщик полетов»	Автор. свид	Свидетельство о внесении сведений в государственный реестр прав на объекты, охраняемые авторским правом № 16867 от «22» апреля 2021 года.	1	Ж.Ж. Айнакулов, Г.Е. Курманкулова.
4	Разработка комплекса интерактивных обучающих программ по технологическим процессам ремонта авиационной техники.	Автор. свид	Свидетельство о внесении сведений в государственный реестр прав на объекты, охраняемые авторским правом № 28907 от «20» сентября 2022 года.	1	Астапенко Н.В., Пирманов И.А., Федоров И.О., Кошеков К.Т., Сейдахметов Б.К., Зуев Д.В., Айнакулов, Ж.Ж.

Соискатель

Список подтверждаю

Ученый секретарь



Ж.Ж. Айнакулов

Г.М. Дерипсалдина

5	Математический алгоритм оценки риска специалиста по техническому обслуживанию воздушных судов в моделировании VR-тренинга.	Автор. свид	Свидетельство о внесении сведений в государственный реестр прав на объекты, охраняемые авторским правом № 28908 от «20» сентября 2022 года.	1	Астапенко Н.В., Пирманов И.А., Федоров И.О., Кошечков К.Т., Сейдахметов Б.К., Зуев Д.В., Айнакулов, Ж.Ж.
6	Разработка цифровой обучающей модели (ЦОМ) для ремонта вертолетной техники.	Автор. свид	Свидетельство о внесении сведений в государственный реестр прав на объекты, охраняемые авторским правом № 29068 от «27» сентября 2022 года.	1	Астапенко Н.В., Пирманов И.А., Федоров И.О., Кошечков К.Т., Зуев Д.В., Айнакулов, Ж.Ж.
7	Программный комплекс управления средствами радиоэлектронной разведки на базе SDR и БПЛА	Автор. свид	Свидетельство о регистрации программы для ЭВМ № 60241 от 25 июня 2025 г.	1	Доля, А. В., Айнакулов, Ж. Ж., Улаков, Е. Т., Бебенин, А. А., Карибаев, Б. А., Кулиш, М. И., Тулембаев, А. Н., & Кажыбаев, К. С.

Соискатель

Список подтверждаю

Ученый секретарь



Ж.Ж. Айнакулов

Г.М. Дерипсалдина

23.04.2026.



АКТ ВНЕДРЕНИЯ
научно-технической продукции
№ 03-И-03/3 от 30 января 2017г.

1. Наименование научно-технической продукции: технологический комплекс «Мобильный комплекс дистанционного зондирования на основе беспилотного летательного аппарата».

2. Вид документа: Описание порядка осуществления аэрофотосъемочных работ с БПЛА.

3. Заказчик научной, научно-технической продукции: Аэрокосмический комитет Министерства обороны и аэрокосмической промышленности Республики Казахстан.

4. Исполнитель работ: АО «Национальный центр космических исследований и технологий».

5. Основание для разработки научной продукции (разработки нового технического средства): Договор с Аэрокосмическим комитетом МОАП РК на выполнение научно-исследовательских работ по Республиканской бюджетной программе 008 «Прикладные научные исследования в области космической деятельности» № 14 от 1 февраля 2017 г. дополнительное соглашение № 1 от 15 марта 2017 г.

6. Сведения о внедрении: Технология выполнения аэрофотосъемки с использованием Беспилотных Летательных Аппаратов предназначена для выполнения крупномасштабного картографирования территорий промышленного и иного назначения. С помощью данной технологии обеспечивается оперативный реалистический мониторинг процессов происходящих на исследуемой территории. Технология предусматривает получение следующих данных:

- файл рельефа местности снабженный изолиниями равной высоты с шагом от 0,5 метра;
- ортофотоплан местности с разрешением 3 – 5 см/пикс в зависимости от высоты съемки (50 – 200 м);
- объемов выработок и отвалов;
- реалистичной трехмерной модели местности с возможностью ее распечатки на 3D-принтере;
- векторные карты содержащие необходимое число слоев с сопутствующей семантической информацией;
- обеспечение возможности построения геоинформационной базы.

Обработка материалов съемки послужила хорошей методической основой при изучении фотограмметрического процесса обработки данных с БПЛА.

7. Сведения об эффективности внедрения (планируемый эффект, срок полезного использования)

Планируемый эффект - технология позволит организовать оперативное обеспечение достоверной информацией о производительности и эффективности разрабатываемых карьеров и промышленных территорий, а также окажет существенную помощь а планировании дальнейших работ по объемам извлекаемого грунта и породы.

8. Сведения о защите интеллектуальной собственности (наименование патента/заявки, регистрационный номер патента/заявки): не имеется

Научный руководитель:

Байгурии

Ж.Д. Байгурии

Структура проекта

bash

neuro_scout_drone/

Подключение RTSP-камеры

```
# camera_stream.py
```

```
import cv2
```

```
def get_rtsp_stream(rtsp_url):
```

```
    return cv2.VideoCapture(rtsp_url)
```

```
# Пример использования:
```

```
# stream = get_rtsp_stream("rtsp://user:pass@ip:554/stream1")
```

Проанализировано

python

```
from graphviz import Digraph
```

```
# Создание диаграммы
```

```
dot = Digraph(format='png')
```

```
dot.attr(bgcolor='white', rankdir='TB', style='filled', fillcolor='azure1',
fontname='Arial')
```

```
# Узлы
```

```
dot.node('A', 'Запуск потока с камеры\n(get_rtsp_stream)', shape='box',
style='filled', fillcolor='paleturquoise')
```

```
dot.node('B', 'Получение кадра\nret, frame = cap.read()', shape='box', style='filled',
fillcolor='paleturquoise')
```

```
dot.node('C', 'Если кадр не получен?', shape='diamond', style='filled',
fillcolor='lightcyan')
```

```
dot.node('D', 'continue', shape='box', style='filled', fillcolor='paleturquoise')
```

```
dot.node('E', 'Обнаружение объектов\n(detect)', shape='box', style='filled',
fillcolor='paleturquoise')
```

```
dot.node('F', 'Проверка аномалий текстуры\n(detect_texture_anomaly)',
shape='box', style='filled', fillcolor='paleturquoise')
```

```
dot.node('G', 'Получение координат GPS\n(get_gps_coords)', shape='box',
style='filled', fillcolor='paleturquoise')
```

```
dot.node('H', 'Обнаружены объекты\nили аномалия?', shape='diamond',
style='filled', fillcolor='lightcyan')
```

```
dot.node('I', 'Формирование\nсообщения Telegram', shape='box', style='filled',
fillcolor='paleturquoise')
```

```
dot.node('J', 'Отправка сообщения\n(send_telegram_alert)', shape='box',
style='filled', fillcolor='paleturquoise')
```

```
dot.node('K', 'Управление Gimbal\n(set_gimbal_angle)', shape='box', style='filled',
fillcolor='paleturquoise')
```

```
# Ребра
```

```

dot.edges([( 'A', 'B'), ('B', 'C')])
dot.edge('C', 'D', label='Да')
dot.edge('C', 'E', label='Нет')
dot.edge('E', 'F')
dot.edge('F', 'G')
dot.edge('G', 'H')
dot.edge('H', 'I', label='Да')
dot.edge('H', 'B', label='Нет')
dot.edge('I', 'J')
dot.edge('I', 'K', label='если есть объект')
# Сохранение диаграммы
output_path = '/mnt/data/surveillance_flowchart'
dot.render(output_path)
output_path + '.png'

```

Код на Python + TensorFlow:

```

python
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
model = Sequential([
    LSTM(128, input_shape=(100, 6), return_sequences=False),
    Dense(64, activation='relu'),
    Dense(7) # x, y, z, qw, qx, qy, qz
])
model.compile(optimizer='adam', loss='mse')
model.summary()

```

Обучение используется синхронизированный набор данных: IMU + GPS/RTK + ориентация с визуального одометра.

Потери: среднеквадратичная ошибка (MSE) по координатам и углам.

Применение сверточных сетей для визуальной ориентации

В случае наличия RGB-камеры, возможно обучение модели по изображению (PoseNet, DeepVO и т.д.), где:

- Вход: изображение или последовательность кадров
- Выход: позиция и ориентация

Пример архитектуры на базе CNN:

```

python
# Используем CNN + регрессию
model = tf.keras.Sequential([
    tf.keras.layers.Conv2D(32, 3, activation='relu', input_shape=(128, 128, 3)),
    tf.keras.layers.MaxPooling2D(),

```

```

tf.keras.layers.Conv2D(64, 3, activation='relu'),
tf.keras.layers.Flatten(),
tf.keras.layers.Dense(128, activation='relu'),
tf.keras.layers.Dense(7) # [x, y, z, qw, qx, qy, qz]
])
import tkinter as tk
from tkinter import filedialog
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import quaternion
import json
import threading
# Визуализация ориентации
class OrientationViewer:
    def __init__(self):
        self.root = tk.Tk()
        self.root.title("Ориентация БПЛА")
        self.root.geometry("300x150")
        self.button_load = tk.Button(self.root, text="Загрузить ориентацию",
command=self.load_orientation)
        self.button_load.pack(pady=20)
        self.status = tk.Label(self.root, text="Файл не загружен", fg="gray")
        self.status.pack()
        self.root.mainloop()
    def load_orientation(self):
        file_path = filedialog.askopenfilename(filetypes=[("JSON Files", "*.json")])
        if file_path:
            try:
                with open(file_path, 'r') as f:
                    data = json.load(f)
                    orientation_sequence = data["quaternions"] # список [qw, qx, qy, qz]
                    self.status.config(text="Анимация...")
                    threading.Thread(target=self.animate_orientation,
args=(orientation_sequence,)).start()
            except Exception as e:
                self.status.config(text=f"Ошибка: {e}")
    def animate_orientation(self, sequence):
        fig = plt.figure()
        ax = fig.add_subplot(111, projection='3d')
        ax.set_xlim([-1, 1])

```

```

ax.set_ylim([-1, 1])
ax.set_zlim([-1, 1])
for quat in sequence:
    ax.cla()
    ax.set_xlim([-1, 1])
    ax.set_ylim([-1, 1])
ax.set_zlim([-1, 1])
    ax.set_xlabel('X')
    ax.set_ylabel('Y')
    ax.set_zlabel('Z')
    ax.set_title('Анимация ориентации БПЛА')
    q = np.quaternion(quat[0], quat[1], quat[2], quat[3])
    v_x = quaternion.rotate_vectors(q, np.array([[1, 0, 0]]))[0]
    v_y = quaternion.rotate_vectors(q, np.array([[0, 1, 0]]))[0]
    v_z = quaternion.rotate_vectors(q, np.array([[0, 0, 1]]))[0]
    ax.quiver(0, 0, 0, v_x[0], v_x[1], v_x[2], color='r', length=1, label='X')
    ax.quiver(0, 0, 0, v_y[0], v_y[1], v_y[2], color='g', length=1, label='Y')
    ax.quiver(0, 0, 0, v_z[0], v_z[1], v_z[2], color='b', length=1, label='Z')
plt.pause(0.1)
plt.show()

```

```

if __name__ == "__main__":
    OrientationViewer()
    // WebGL / Three.js сцена визуализации маршрута и ориентации БПЛА
    // Зависимости: three.js
    import * as THREE from 'three';
    import { OrbitControls } from 'three/examples/jsm/controls/OrbitControls';
    import { GLTFLoader } from 'three/examples/jsm/loaders/GLTFLoader';
    let scene, camera, renderer, drone;
    let gpsPositions = []; // массив [{lat, lon, alt}]
    let quaternions = []; // массив [x, y, z, w]
    let index = 0;
    let clock = new THREE.Clock();
    function init() {
        scene = new THREE.Scene();
        scene.background = new THREE.Color(0xbfd1e5);
        camera = new THREE.PerspectiveCamera(75,
window.innerWidth/window.innerHeight, 0.1, 1000);
        camera.position.set(0, 20, 50);
        renderer = new THREE.WebGLRenderer({ antialias: true });
        renderer.setSize(window.innerWidth, window.innerHeight);
    }

```

```

document.body.appendChild(renderer.domElement);
const controls = new OrbitControls(camera, renderer.domElement);
controls.update();
const light = new THREE.DirectionalLight(0xffffff, 1);
light.position.set(10, 20, 10);
scene.add(light);
const grid = new THREE.GridHelper(200, 20);
scene.add(grid);
// Загрузка 3D модели дрона (можно заменить на BoxGeometry)
const loader = new GLTFLoader();
loader.load('drone_model.glb', function(gltf) {
    drone = gltf.scene;
    drone.scale.set(2, 2, 2);
    scene.add(drone);
});
animate();
}
function loadFlightData(jsonData) {
    gpsPositions = jsonData.gps.map(([lat, lon, alt]) => new THREE.Vector3(lon *
100, alt, lat * 100));
    quaternions = jsonData.quaternions.map([w, x, y, z]) => new
THREE.Quaternion(x, y, z, w));
}
function animate() {
    requestAnimationFrame(animate);
    if (drone && gpsPositions.length > 0) {
        let i = Math.floor(clock.getElapsedTime() * 2) % gpsPositions.length;
        drone.position.copy(gpsPositions[i]);
        drone.quaternion.copy(quaternions[i]);
    }
    renderer.render(scene, camera);
}

// Пример загрузки данных из JSON
fetch('flight_data.json')
    .then(response => response.json())
    .then(data => {
        loadFlightData(data);
    });
init();

```

Модель оценки ориентации БПЛА на основе IMU (TensorFlow Lite)

Подготовка входных данных

Типичные данные IMU (вектор из 6 параметров):

LSTM-модель оценки положения и ориентации БПЛА

Алгоритм LSTM_Pose_Estimation. Вход:

$X \leftarrow$ временная последовательность длины T (IMU, GPS/RTK, визуальная одометрия). Размерность: $T \times 6$. Выход:

$Y \leftarrow [x, y, z, qw, qx, qy, qz]$

Инициализация:

Создать модель LSTM(128 нейронов), Dense(64, функция активации ReLU), Dense(7)

Задаем оптимизатор Adam, функцию потерь: MSE. Обучение для каждой эпохи и для каждого батча данных:

Прямой проход (Forward pass)

Вычислить ошибку (MSE) и обновить веса (Backpropagation Through Time). В результате получим обученную модель прогнозирования положения и ориентации.

CNN-модель визуальной ориентации БПЛА

Алгоритм CNN_Visual_Pose, на входе:

$I \leftarrow$ RGB-изображение или последовательность кадров с размерностью:
 $128 \times 128 \times 3$

На выходе:

$Pose \leftarrow [x, y, z, qw, qx, qy, qz]$

Инициализация. Создается CNN

Conv(32) - ReLU - MaxPooling

Conv(64) - ReLU

Flatten

Dense(128) - ReLU

Dense(7)

Функция потерь: MSE

Оптимизатор: Adam

Обучение:

Для каждой эпохи: прямой проход, вычисление ошибки и обновление весов.

Результат: Оценка позиции и ориентации по визуальным данным.

Интеграция в интеллектуальную систему БПЛА

IMU, GPS и VO - LSTM - прогноз состояния

RGB-камера - CNN - визуальная ориентация

Объединение - EKF / RL / MPC - автопилот

Рассматриваемый код преобразует кватернион во вращение трех осей, рисует стрелки в 3D: красная - X (roll), зеленая - Y (pitch), синяя - Z (yaw).

Это позволяет наглядно видеть, как дрон ориентирован в пространстве.

Опционально можно добавить анимацию в `matplotlib.animation`, добавить поддержку MAVLink потока с ориентацией и визуализировать путь движения по GPS и ориентации на карте или DEM.

При интеграции необходимо для Jetson: запускать этот код по результатам TFLite-модели, для Unity / Unreal: передавать кватернионы через сокет и крутить модель, для Web: использовать `three.js` и `WebSocket` и передавать ориентацию с микроконтроллера (Приложение В).

Визуализация ориентации БПЛА (Desktop GUI). Назначение: загрузка и анимация ориентации БПЛА, заданной кватернионами.

Алгоритм `Orientation_View`. Инициализация:

Создать графическое окно. Добавить кнопку "Загрузить ориентацию"

Добавить строку состояния. Обработка события "Загрузить" и открыть диалог выбора файла. Если файл выбран, то считать JSON-данные и извлечь последовательность кватернионов. Запустить анимацию в отдельном потоке, где анимация ориентации создание 3D-графика.

Для каждого кватерниона в последовательности выполняется: очистить сцену, преобразовать базисные оси (X, Y, Z) кватернионом и отобразить повернутые оси в 3D. Сделать паузу (dt).

Результатом является анимированное отображение ориентации БПЛА.

Веб-визуализация маршрута и ориентации БПЛА (WebGL / Three.js)

Назначение: Интерактивная 3D-визуализация траектории и ориентации БПЛА в браузере. Алгоритм `WebGL_UAV_Visualization`. Инициализация сцены создать 3D-сцену, камеру и рендер. Добавить освещение и координатную сетку, подключить управление камерой, загрузить 3D-модель БПЛА.

Загрузка полетных данных путем считывания JSON-файлов, преобразования GPS-координаты в 3D-точки, преобразования кватернионы в формат 3D-движка. Основной цикл анимации:

Пока сцена активна надо определить текущий индекс по времени, установить позицию и ориентацию БПЛА. Отрисовать сцену

Результат: 3D-анимация движения и ориентации БПЛА в реальном времени. Обобщенная архитектура визуализации (Приложение Г).

JSON (GPS и Quaternion)



Предобработка данных



Desktop GUI (Matplotlib) | WebGL (Three.js)



3D-анимация ориентации и траектории БПЛА

`flight_data.json` - пример GPS и ориентационных данных.

Пример кода на Python (TensorFlow / Keras)

```
python
import numpy as np
import rasterio
from sklearn.preprocessing import MinMaxScaler
# Чтение DEM и NDVI (пример - 3 временных среза)
dem_paths = ["dem_t1.tif", "dem_t2.tif", "dem_t3.tif"]
ndvi_paths = ["ndvi_t1.tif", "ndvi_t2.tif", "ndvi_t3.tif"]
def load_raster(path):
    with rasterio.open(path) as src:
        return src.read(1)
dem_series = [load_raster(p) for p in dem_paths]
ndvi_series = [load_raster(p) for p in ndvi_paths]
# Преобразование в numpy массив [time, height, width, channels]
X = np.stack([np.stack([dem, ndvi], axis=-1) for dem, ndvi in zip(dem_series,
ndvi_series)])
X = X.astype(np.float32)
# Прогноз на 1 шаг вперед: прогноз высоты DEM
Y = np.stack(dem_series[1:], axis=0) # target heights
# Обрезка X, чтобы соответствовал Y по времени
X = X[:-1]
# Масштабирование
scaler = MinMaxScaler()
X_flat = X.reshape(-1, 2)
X_scaled = scaler.fit_transform(X_flat).reshape(X.shape)
Y_scaled = scaler.fit_transform(Y.reshape(-1, 1)).reshape(Y.shape)
Построение LSTM-модели
python
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, TimeDistributed, Flatten, Reshape
# reshape X: [samples, timesteps, features]
X_lstm = X_scaled.reshape((X_scaled.shape[0], -1, X_scaled.shape[-1])) # e.g., [2,
H*W, 2]
Y_lstm = Y_scaled.reshape((Y_scaled.shape[0], -1, 1))
model = Sequential()
model.add(LSTM(64, input_shape=(X_lstm.shape[1], X_lstm.shape[2]),
return_sequences=True))
model.add(TimeDistributed(Dense(1)))
model.compile(optimizer='adam', loss='mse')
model.fit(X_lstm, Y_lstm, epochs=20, batch_size=1)
```

Прогноз и визуализация

python

```
import matplotlib.pyplot as plt
```

```
# Предсказание
```

```
predicted = model.predict(X_lstm)
```

```
predicted_resaped = predicted.reshape(Y.shape)
```

```
# Сравнение с истинными данными
```

```
plt.figure(figsize=(12,4))
```

```
plt.subplot(1, 2, 1)
```

```
plt.title("Настоящий DEM (t+1)")
```

```
plt.imshow(Y[0], cmap='terrain')
```

```
plt.colorbar()
```

```
plt.subplot(1, 2, 2)
```

```
plt.title("Прогнозируемый DEM (t+1)")
```

```
plt.imshow(predicted_resaped[0], cmap='terrain')
```

```
plt.colorbar()
```

```
plt.tight_layout()
```

```
plt.show()
```

Структура входных данных (предполагаемая)

- data/dem_YYYYMMDD.tif - цифровые модели рельефа.
- data/ndvi_YYYYMMDD.tif - NDVI с мультиспектральной камеры или Sentinel-2.
- Формат: GeoTIFF, одна полоса (1-band).

Установка библиотек

bash

```
pip install rasterio numpy scikit-learn tensorflow matplotlib
```

Загрузка и подготовка GeoTIFF-данных

python

```
import os
```

```
import numpy as np
```

```
import rasterio
```

```
from glob import glob
```

```
from sklearn.preprocessing import MinMaxScaler
```

```
# Загрузка GeoTIFF по шаблону
```

```
def load_stack(pattern):
```

```
    paths = sorted(glob(pattern))
```

```
    arrays = []
```

```
    for p in paths:
```

```
        with rasterio.open(p) as src:
```

```
            arrays.append(src.read(1))
```

```
    return np.stack(arrays), src.profile # Вернем и профиль GeoTIFF
```

Пути к данным

```
dem_stack, dem_profile = load_stack("data/dem_*.tif")
```

```
ndvi_stack, _ = load_stack("data/ndvi_*.tif")
```

```
print("DEM shape:", dem_stack.shape) # (time, H, W)
```

1. detect_and_analyze.py

YOLOv8 модель

```
python
```

```
model = YOLO("yolov8n.pt")
```

Загружается облегченная нейросеть YOLOv8n, подходящая для real-time инференса на Jetson Nano / Xavier NX.

Обнаружение объектов

```
python
```

```
results = model(frame)[0]
```

YOLO возвращает предсказания: ограничивающие рамки (хуху), классы объектов и уверенность (confidence). Выход:

```
python
```

```
[{'bbox': (x1, y1, x2, y2), 'class_id': 0, 'confidence': 0.91}, ...]
```

Детекция аномалий текстуры

```
python
```

```
stddev = np.std(gray)
```

```
return stddev > threshold
```

Если стандартное отклонение яркости в кадре превышает порог - рельеф/текстура считается аномальной (например, резкое изменение цвета или структуры, как у обрушений, пятен, следов).

2. camera_stream.py

Подключение к видеопотоку через RTSP:

```
python
```

```
cv2.VideoCapture(rtsp_url)
```

Поддерживает ONVIF-камеры и IP-камеры.

3. gps_tracker.py

Чтение координат с GPS по UART (/dev/ttyTHS1). NMEA-строка \$GPGGA:

```
text
```

```
$GPGGA,123519,4807.038,N,01131.000,E,...
```

Координаты извлекаются в виде широты и долготы.

4. gimbal_control.py

Связь с автопилотом через MAVLink:

```
python
```

```
master = mavutil.mavlink_connection(...)
```

Команда поворота камеры:

```
python
```

```
mavutil.mavlink.MAV_CMD_DO_MOUNT_CONTROL
```

Передаются углы наклона (pitch) и поворота (yaw). Угол yaw задается пропорционально координате обнаруженного объекта в кадре.

5. alert_sender.py

Отправка Telegram-уведомлений:

```
python
```

```
https://api.telegram.org/bot<BOT_TOKEN>/sendMessage
```

Сообщение включает координаты, ID объектов и предупреждение об аномалии.

6. main_loop.py – Главный цикл

- Читает кадр.

- Вызывает detect() и detect_texture_anomaly().

- Получает координаты GPS.

- Отправляет Telegram-оповещение, если есть события.

- Управляет Gimbal на основе первого обнаруженного объекта.

Нормализация

```
scaler = MinMaxScaler()
```

```
def scale_cube(cube):
```

```
    flat = cube.reshape(-1, 1)
```

```
    scaled = scaler.fit_transform(flat)
```

```
    return scaled.reshape(cube.shape)
```

```
dem_scaled = scale_cube(dem_stack)
```

```
ndvi_scaled = scale_cube(ndvi_stack)
```

```
# Объединение в единый тензор: [time, H, W, channels]
```

```
X = np.stack([dem_scaled[:,-1], ndvi_scaled[:,-1]], axis=-1).transpose(1, 2, 3, 0)
```

```
X = X.transpose(3, 1, 2, 0) # [time, H, W, channels]
```

```
Y = dem_scaled[1:] # следующий шаг DEM
```

Подготовка данных для LSTM

```
# [time, H, W, channels] => [samples, timesteps, features]
```

```
t, h, w, c = X.shape
```

```
X_lstm = X.reshape(t, -1, c) # [samples, H*W, 2]
```

```
Y_lstm = Y.reshape(t, -1, 1) # [samples, H*W, 1]
```

Обучение LSTM

```
from tensorflow.keras.models import Sequential
```

```
from tensorflow.keras.layers import LSTM, TimeDistributed, Dense
```

```
model = Sequential()
```

```
model.add(LSTM(64, input_shape=(X_lstm.shape[1], X_lstm.shape[2]),  
return_sequences=True))
```

```
model.add(TimeDistributed(Dense(1)))
```

```
model.compile(optimizer='adam', loss='mse')
```

```
model.fit(X_lstm[:,-1], Y_lstm[1:], epochs=10, batch_size=1)
```

Прогноз и визуализация в 3D

```
from mpl_toolkits.mplot3d import Axes3D
```

```

import matplotlib.pyplot as plt
    predicted = model.predict(X_lstm[-1:]) # Последний фрейм
pred = predicted[0].reshape(h, w)
# Обратное масштабирование (если нужно)
pred_orig = scaler.inverse_transform(pred.reshape(-1, 1)).reshape(h, w)
# 3D-визуализация
x, y = np.meshgrid(np.arange(w), np.arange(h))
fig = plt.figure(figsize=(12, 6))
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface(x, y, pred_orig, cmap='terrain', linewidth=0, antialiased=False)
ax.set_title("Прогнозируемая модель рельефа (DEM t+1)")
ax.set_xlabel("X")
ax.set_ylabel("Y")
ax.set_zlabel("Высота (м)"/plt.tight_layout()/plt.show()

```

Пример Python-кода для интеграции

```
python
import numpy as np
import torch
import torch.nn as nn
import rasterio
from rasterio.plot import show

Загрузка NDVI-данных ---
with rasterio.open('sentinel_ndvi.tif') as src:
    ndvi_data = src.read(1)
Нормализация данных ---
ndvi_tensor = torch.tensor(ndvi_data,
dtype=torch.float32).unsqueeze(0).unsqueeze(0)
ndvi_tensor = (ndvi_tensor - ndvi_tensor.min()) / (ndvi_tensor.max() -
ndvi_tensor.min())
Определение гибридной модели ---
class CNN_LSTM(nn.Module):
    def __init__(self):
        super(CNN_LSTM, self).__init__()
        self.cnn = nn.Sequential(
            nn.Conv2d(1, 16, 3, padding=1),
            nn.ReLU(),
            nn.MaxPool2d(2),
            nn.Conv2d(16, 32, 3, padding=1),
            nn.ReLU(),
            nn.MaxPool2d(2)
        )
        self.lstm = nn.LSTM(32*64*64, 128, batch_first=True)
        self.fc = nn.Linear(128, 1)
    def forward(self, x_seq):
        batch_size, seq_len, c, h, w = x_seq.size()
        cnn_out = []
        for t in range(seq_len):
            x = self.cnn(x_seq[:, t, :, :])
            cnn_out.append(x.view(batch_size, -1))
        cnn_seq = torch.stack(cnn_out, dim=1)
        lstm_out, _ = self.lstm(cnn_seq)
        return self.fc(lstm_out[:, -1, :])
Пример данных (последовательность NDVI-карт) ---
sequence = torch.randn(1, 5, 1, 256, 256)
```

```

model = CNN_LSTM()
output = model(sequence)
print("Прогноз состояния объекта:", output.item())

```

Применение в ГИС-среде. Интеграция осуществляется через:

- ****GeoServer / PostGIS**** - хранение и публикация прогнозных карт;
- ****QGIS / ArcGIS**** - визуализация результатов и анализ пространственных изменений;
- ****API-интерфейсы (REST / OGC WMS/WFS)**** - автоматическая передача прогнозов и обратной связи в интеллектуальный модуль.

Детекция объектов и аномалий

Структура проекта

bash

neuro_scout_drone/

Подключение RTSP-камеры

camera_stream.py

import cv2

def get_rtsp_stream(rtsp_url):

return cv2.VideoCapture(rtsp_url)

Пример использования:

stream = get_rtsp_stream("rtsp://user:pass@ip:554/stream1")

Проанализировано

python

from graphviz import Digraph

Создание диаграммы

dot = Digraph(format='png')

dot.attr(bgcolor='white', rankdir='TB', style='filled', fillcolor='azure1', fontname='Arial')

Узлы

dot.node('A', 'Запуск потока с камеры\n(get_rtsp_stream)', shape='box', style='filled', fillcolor='paleturquoise')

dot.node('B', 'Получение кадра\nret, frame = cap.read()', shape='box', style='filled', fillcolor='paleturquoise')

dot.node('C', 'Если кадр не получен?', shape='diamond', style='filled', fillcolor='lightcyan')

dot.node('D', 'continue', shape='box', style='filled', fillcolor='paleturquoise')

dot.node('E', 'Обнаружение объектов\n(detect)', shape='box', style='filled', fillcolor='paleturquoise')

dot.node('F', 'Проверка аномалий текстуры\n(detect_texture_anomaly)', shape='box', style='filled', fillcolor='paleturquoise')

```

dot.node('G', 'Получение координат GPS\n(get_gps_coords)', shape='box',
style='filled', fillcolor='paleturquoise')
dot.node('H', 'Обнаружены объекты\nили аномалия?', shape='diamond',
style='filled', fillcolor='lightcyan')
dot.node('I', 'Формирование\nсообщения Telegram', shape='box', style='filled',
fillcolor='paleturquoise')
dot.node('J', 'Отправка сообщения\n(send_telegram_alert)', shape='box',
style='filled', fillcolor='paleturquoise')
dot.node('K', 'Управление Gimbal\n(set_gimbal_angle)', shape='box', style='filled',
fillcolor='paleturquoise')
# Ребра
dot.edges([( 'A', 'B'), ('B', 'C')])
dot.edge('C', 'D', label='Да')
dot.edge('C', 'E', label='Нет')
dot.edge('E', 'F')
dot.edge('F', 'G')
dot.edge('G', 'H')
dot.edge('H', 'I', label='Да')
dot.edge('H', 'B', label='Нет')
dot.edge('I', 'J')
dot.edge('I', 'K', label='если есть объект')
# Сохранение диаграммы
output_path = '/mnt/data/surveillance_flowchart'
dot.render(output_path)
output_path + '.png'

```

Принцип работы визуализации маршрута БПЛА, его ориентации и поверхности рельефа (DEM) с возможностью экспорта сцены в формате .glb с использованием Three.js:

Шаг 1. Подключение зависимостей

```

js
import * as THREE from 'three';
import { OrbitControls } from 'three/examples/jsm/controls/OrbitControls';
import { GLTFExporter } from 'three/examples/jsm/exporters/GLTFExporter';
- THREE.js - рендеринг 3D-сцены.
- OrbitControls - управление камерой мышью.
- GLTFExporter - экспорт в .glb.

```

Шаг 2. Создание сцены

```

js
scene = new THREE.Scene();

```

```
scene.background = new THREE.Color(0xbfd1e5);
```

Задаем цвет фона и создаем сцену.

Шаг 3. Камера и рендерер

```
js
```

```
camera = new THREE.PerspectiveCamera(75,  
window.innerWidth/window.innerHeight, 0.1, 1000);  
camera.position.set(0, 20, 50);  
renderer = new THREE.WebGLRenderer({ antialias: true });  
renderer.setSize(window.innerWidth, window.innerHeight);  
document.body.appendChild(renderer.domElement);
```

- Камера "от первого лица", движется за дроном.

- Рендерим в браузер (canvas в <body>).

Шаг 4. Свет и навигация

```
js
```

```
const controls = new OrbitControls(camera, renderer.domElement);  
const light = new THREE.DirectionalLight(0xffffff, 1);  
light.position.set(10, 20, 10);  
scene.add(light);  
const grid = new THREE.GridHelper(200, 20);  
scene.add(grid);
```

Направленный свет и вспомогательная сетка.

Шаг 5. Модель БПЛА (временно - Box)

```
js
```

```
const geometry = new THREE.BoxGeometry(2, 1, 3);  
const material = new THREE.MeshStandardMaterial({ color: 0xff5533 });  
drone = new THREE.Mesh(geometry, material);  
scene.add(drone);
```

Пока используется BoxGeometry, его можно заменить на .glb или .obj модель.

Шаг 6. Поверхность DEM

```
js
```

```
const demGeo = new THREE.PlaneGeometry(400, 400, 20, 20);  
const demMat = new THREE.MeshStandardMaterial({ color: 0x8ac926, wireframe:  
true });  
const demMesh = new THREE.Mesh(demGeo, demMat);  
demMesh.rotation.x = -Math.PI / 2;  
scene.add(demMesh);
```

Плоская поверхность с имитацией рельефа. Можно заменить на реальные данные DEM, если будет высотная карта.

Шаг 7. Загрузка GPS и кватернионов

```
js
```

```
function loadFlightData(jsonData) {
```

```

    gpsPositions = jsonData.gps.map(([lat, lon, alt]) => new THREE.Vector3(lon *
100, alt, lat * 100));
    quaternions = jsonData.quaternions.map([w, x, y, z]) => new
THREE.Quaternion(x, y, z, w));
}

```

- Преобразуем координаты в THREE.Vector3.
- Кватернионы - повороты модели.

Шаг 8. Отрисовка маршрута

```

js
const pathMat = new THREE.LineBasicMaterial({ color: 0x0000ff });
const pathGeo = new THREE.BufferGeometry().setFromPoints(gpsPositions);
const flightPath = new THREE.Line(pathGeo, pathMat);
scene.add(flightPath);

```

Визуализируем полет в виде синей линии.

Шаг 9. Анимация полета

```

js
function animate() {
    requestAnimationFrame(animate);
    if (drone && gpsPositions.length > 0) {
        let i = Math.floor(clock.getElapsedTime() * 2) % gpsPositions.length;
        drone.position.copy(gpsPositions[i]);
        drone.quaternion.copy(quaternions[i]);
        camera.position.lerp(drone.position.clone().add(new THREE.Vector3(0, 5, -
10).applyQuaternion(drone.quaternion)), 0.05);
        camera.lookAt(drone.position);
    }
    renderer.render(scene, camera);
}

```

- Дрон "летит" по маршруту с анимацией ориентации.
- Камера следует за дроном.

Шаг 10. Кнопка экспорта

```

js
function exportSceneToGLB() {
    const exporter = new GLTFExporter();
    exporter.parse(
        scene,
        function (gltf) {
            const blob = new Blob([gltf], { type: 'application/octet-stream' });
            const link = document.createElement('a');
            link.href = URL.createObjectURL(blob);
            link.download = 'drone_scene.glb';
        }
    );
}

```

```

        link.click();
    },
    { binary: true }
);
}

```

Экспорт всей сцены как .glb с анимацией и объектами.

Шаг 11. Интерфейс кнопки

```

js
function createExportButton() {
    exportButton = document.createElement('button');
    exportButton.innerText = 'Экспортировать сцену';
    exportButton.style.position = 'absolute';
    exportButton.style.top = '20px';
    exportButton.style.left = '20px';
    exportButton.style.padding = '10px';
    exportButton.style.zIndex = '1';
    document.body.appendChild(exportButton);
    exportButton.onclick = exportSceneToGLB;
}

```

Шаг 12. Загрузка flight_data.json

```

js
fetch('flight_data.json')
    .then(response => response.json())
    .then(data => {
        loadFlightData(data);
    });

```

Формат файла:

```

json
{
    "gps": [[lat, lon, alt], ...],
    "quaternions": [[w, x, y, z], ...]
}

```

перенести визуализацию ориентации БПЛА с tkinter и matplotlib в полноценную WebGL-сцену на сайте с Three.js. Нужно объединить код в структуру, которую можно использовать в веб-приложении.

Протестировать локально или встроить на сайт:

```

index.html
html
<!DOCTYPE html>
<html lang="en">
<head>

```

```

<meta charset="UTF-8" />
<title>Визуализация ориентации БПЛА</title>
<style>
  body { margin: 0; overflow: hidden; }
  canvas { display: block; }
</style>
</head>
<body>

<script type="module">
import * as THREE from
'https://cdn.jsdelivr.net/npm/three@0.160.1/build/three.module.js';
import { OrbitControls } from
'https://cdn.jsdelivr.net/npm/three@0.160.1/examples/jsm/controls/OrbitControls.js';
import { GLTFLoader } from
'https://cdn.jsdelivr.net/npm/three@0.160.1/examples/jsm/loaders/GLTFLoader.js';

let scene, camera, renderer, drone;
let gpsPositions = [];
let quaternions = [];
let clock = new THREE.Clock();

init();

function init() {
  scene = new THREE.Scene();
  scene.background = new THREE.Color(0xbfd1e5);
  camera = new THREE.PerspectiveCamera(75,
window.innerWidth/window.innerHeight, 0.1, 1000);
  camera.position.set(0, 20, 50);
  renderer = new THREE.WebGLRenderer({ antialias: true });
  renderer.setSize(window.innerWidth, window.innerHeight);
  document.body.appendChild(renderer.domElement);
  const controls = new OrbitControls(camera, renderer.domElement);
  controls.update();
  const light = new THREE.DirectionalLight(0xffffff, 1);
  light.position.set(10, 20, 10);
  scene.add(light);
  const grid = new THREE.GridHelper(200, 20);
  scene.add(grid);
  // Загрузка 3D модели дрона

```

```

const loader = new GLTFLoader();
loader.load('drone_model.glb', function(gltf) {
  drone = gltf.scene;
  drone.scale.set(2, 2, 2);
  scene.add(drone);
}, undefined, function(error) {
  console.error("Ошибка загрузки модели: ", error);
  // Временный куб вместо дрона
  const geometry = new THREE.BoxGeometry(5, 2, 5);
  const material = new THREE.MeshStandardMaterial({ color: 0xff0000 });
  drone = new THREE.Mesh(geometry, material);
  scene.add(drone);
});

// Загрузка данных
fetch('flight_data.json')
  .then(response => response.json())
  .then(data => {
    loadFlightData(data);
  });
animate();
}
function loadFlightData(data) {
  gpsPositions = data.gps.map(([lat, lon, alt]) => new THREE.Vector3(lon * 100, alt,
lat * 100));
  quaternions = data.quaternions.map(([w, x, y, z]) => new THREE.Quaternion(x, y,
z, w));
}
function animate() {
  requestAnimationFrame(animate);
  if (drone && gpsPositions.length > 0) {
    let i = Math.floor(clock.getElapsedTime() * 2) % gpsPositions.length;
    drone.position.copy(gpsPositions[i]);
    drone.quaternion.copy(quaternions[i]);
  }
  renderer.render(scene, camera);
}
</script>
</body>
</html>
flight_data.json (пример)

```

json

КопироватьРедактировать

```
{
  "gps": [
    [43.25, 76.92, 120],
    [43.251, 76.921, 122],
    [43.252, 76.922, 123]
  ],
  "quaternions": [
    [1, 0, 0, 0],
    [0.92388, 0.38268, 0, 0],
    [0.7071, 0.7071, 0, 0]
  ]
}
```

Если нет drone_model.glb:

Ты можешь либо:

Заменить его на BoxGeometry (см. onError в коде),

Или загрузить любую .glb модель дрона и положить в ту же папку.

Как запустить:

1. Создать папку и положить туда index.html, flight_data.json, (и drone_model.glb, если есть).
2. Открыть терминал в этой папке.
3. Запустить простой сервер:

bash

python3 -m http.server

Перейди в браузере на: <http://localhost:8000/>

Внутри:

index.html - WebGL-сцена на Three.js;

Пример сообщения:

less

Координаты: 52.353210, 76.953219

Объекты: [0, 2]

Аномалия рельефа!

Симуляция вывода

Входной кадр:

На изображении обнаружены: человек (class_id=0), автомобиль (class_id=2).

Текстура (например, резко контрастная зона) признана аномальной.

GPS:

makefile

Широта: 52.353210, Долгота: 76.953219

Telegram-уведомление:

less

Событие! Координаты: 52.353210, 76.953219

Объекты: [0, 2]

Аномалия рельефа!

Gimbal:

Камера поворачивается в сторону центра bbox первого объекта (class_id=0), yaw $\approx -15^\circ$.

```

Python-код для интеграции
import numpy as np
import torch
import torch.nn as nn
import rasterio
from rasterio.plot import show
Загрузка NDVI-данных ---
with rasterio.open('sentinel_ndvi.tif') as src:
    ndvi_data = src.read(1)
Нормализация данных ---
ndvi_tensor = torch.tensor(ndvi_data,
dtype=torch.float32).unsqueeze(0).unsqueeze(0)
ndvi_tensor = (ndvi_tensor - ndvi_tensor.min()) / (ndvi_tensor.max() -
ndvi_tensor.min())
Определение гибридной модели ---
class CNN_LSTM(nn.Module):
    def __init__(self):
        super(CNN_LSTM, self).__init__()
        self.cnn = nn.Sequential(
            nn.Conv2d(1, 16, 3, padding=1),
            nn.ReLU(),
            nn.MaxPool2d(2),
            nn.Conv2d(16, 32, 3, padding=1),
            nn.ReLU(),
            nn.MaxPool2d(2)
        )
        self.lstm = nn.LSTM(32*64*64, 128, batch_first=True)
        self.fc = nn.Linear(128, 1)
    def forward(self, x_seq):
        batch_size, seq_len, c, h, w = x_seq.size()
        cnn_out = []
        for t in range(seq_len):
            x = self.cnn(x_seq[:, t, :, :, :])
            cnn_out.append(x.view(batch_size, -1))
        cnn_seq = torch.stack(cnn_out, dim=1)
        lstm_out, _ = self.lstm(cnn_seq)
        return self.fc(lstm_out[:, -1, :])
Пример данных (последовательность NDVI-карт) ---
sequence = torch.randn(1, 5, 1, 256, 256)
model = CNN_LSTM()

```

```

output = model(sequence)
print("Прогноз состояния объекта:", output.item())
1. Подключение к RTSP-камере
python
# camera_stream.py
import cv2
def get_rtsp_stream(rtsp_url):
    return cv2.VideoCapture(rtsp_url)
2. Детекция объектов и аномалий
python
# detect_and_analyze.py
from ultralytics import YOLO
import cv2
import numpy as np
model = YOLO("yolov8n.pt")
def detect(frame):
    results = model(frame)[0]
    objects = []
    for box in results.boxes:
        cls = int(box.cls[0])
        conf = float(box.conf[0])
        x1, y1, x2, y2 = map(int, box.xyxy[0])
        objects.append({
            'bbox': (x1, y1, x2, y2),
            'class_id': cls,
            'confidence': conf
        })
    return objects
def detect_texture_anomaly(frame, threshold=50):
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    stddev = np.std(gray)
    return stddev > threshold
3. Получение координат GPS
python
# gps_tracker.py
import serial
def get_gps_coords(serial_port="/dev/ttyTHS1", baud=9600):
    gps = serial.Serial(serial_port, baud, timeout=1)
    while True:
        line = gps.readline().decode("utf-8", errors="ignore")
        if "$GPGGA" in line:

```

```

    parts = line.split(",")
    if parts[2] and parts[4]:
        lat = float(parts[2]) / 100.0
        lon = float(parts[4]) / 100.0
        return lat, lon

```

4. Управление Gimbal

```

python
# gimbal_control.py
from pymavlink import mavutil
# Подключение к дрону
master = mavutil.mavlink_connection('udpout:127.0.0.1:14550')
master.wait_heartbeat()
def set_gimbal_angle(pitch=0, yaw=0):
    master.mav.command_long_send(
        master.target_system,
        master.target_component,
        mavutil.mavlink.MAV_CMD_DO_MOUNT_CONTROL,
        0,
        pitch, 0, yaw, 0, 0, 0,
        mavutil.mavlink.MAV_MOUNT_MODE_MAVLINK_TARGETING
    )

```

5. Отправка тревог

```

python
# alert_sender.py
import requests
def send_telegram_alert(bot_token, chat_id, message):
    url = f"https://api.telegram.org/bot{bot_token}/sendMessage"
    data = {'chat_id': chat_id, 'text': message}
    requests.post(url, data=data)

```

Основной цикл

```

python

# main_loop.py
from camera_stream import get_rtsp_stream
from detect_and_analyze import detect, detect_texture_anomaly
from gimbal_control import set_gimbal_angle
from gps_tracker import get_gps_coords
from alert_sender import send_telegram_alert
import cv2
RTSP_URL = "rtsp://..."
BOT_TOKEN = "123456:ABC..."

```

```

CHAT_ID = "999999999"
cap = get_rtsp_stream(RTSP_URL)
while True:
    ret, frame = cap.read()
    if not ret:
        continue
    objs = detect(frame)
    anomaly = detect_texture_anomaly(frame)
    lat, lon = get_gps_coords()
    if objs or anomaly:
        message = f" Событие! Координаты: {lat:.6f}, {lon:.6f}"
        if objs:
            message += f"\n Объекты: {[o['class_id'] for o in objs]}"
        if anomaly:
            message += f"\n Обнаружена аномалия рельефа!"
        send_telegram_alert(BOT_TOKEN, CHAT_ID, message)
        if objs:
            center_x = (objs[0]['bbox'][0] + objs[0]['bbox'][2]) / 2
            yaw = (center_x / 640 - 0.5) * 90 # примитивная оценка угла
            set_gimbal_angle(yaw=yaw)

```

Python

```

import numpy as np
import matplotlib.pyplot as plt
# Параметры миссии
waypoints = np.array([[0, 0], [100, 0], [100, 100], [0, 100]])
speed = 10.0 # м/с
dt = 1.0 # шаг симуляции (с)
# Имитация движения БПЛА по точкам
def simulate_flight(waypoints, speed, dt):
    path = []
    shutter_times = []
    t = 0
    for i in range(len(waypoints)-1):
        start = waypoints[i]
        end = waypoints[i+1]
        direction = (end - start)
        distance = np.linalg.norm(direction)
        steps = int(distance / (speed * dt))
        unit_vec = direction / distance
        for s in range(steps):
            position = start + unit_vec * speed * dt * s

```

```

    path.append(position)
    # Предиктивное условие: срабатывание затвора в центре сегмента
    if abs(s - steps//2) < 1:
        shutter_times.append(t)
        t += dt
    return np.array(path), shutter_times
# Моделируем полет
path, shutter_times = simulate_flight(waypoints, speed, dt)
# Отображаем траекторию
plt.figure(figsize=(6, 6))
plt.plot(path[:, 0], path[:, 1], label='Траектория БПЛА')
plt.scatter(*zip(*waypoints), color='red', label='Точки маршрута')
for i, t in enumerate(shutter_times):
    plt.plot(path[t][0], path[t][1], 'bo', label='Срабатывание' if i == 0 else '')
plt.title('Интеллектуальное управление затвором')
plt.xlabel('X (м)')
plt.ylabel('Y (м)')
plt.legend()
plt.grid(True)
plt.show()

```

Объяснение:

БПЛА летит по заданным точкам.

Алгоритм предсказывает момент, когда БПЛА будет в середине отрезка.

В этот момент виртуально срабатывает затвор.

2. Расширение: событие "Обнаружен объект" (YOLO + правило)

python

```

def should_trigger_shutter(obj_detected, distance_to_obj):
    if obj_detected and distance_to_obj < 15:
        return True
    return False

```

Пример вызова:

```
object_detected = True
```

```
distance = 12 # м
```

```
if should_trigger_shutter(object_detected, distance):
```

```
    print("Срабатывание затвора: объект обнаружен на близком расстоянии!")
```

Можно подключить к YOLOv8 через Ultralytics:

bash

```
pip install ultralytics
```

Использовать:

python

```
from ultralytics import YOLO
```

```
model = YOLO("yolov8n.pt")
```

```
results = model('frame.jpg')
```

```
if results[0].boxes:
```

```
    print("Объект обнаружен - активировать затвор")
```

Генетический алгоритм: оптимизация моментов съемки

Если интересует оптимизация выбора времен съемки (по критериям освещенности, четкости, перекрытия):

```
python
```

```
# Псевдокод с DEAP или PyGAD
```

```
# Функция приспособленности:
```

```
def fitness(candidate):
```

```
    score = 0
```

```
    for t in candidate:
```

```
        illum = get_lighting_at(t)
```

```
        sharp = get_camera_sharpness(t)
```

```
        score += illum * 0.6 + sharp * 0.4
```

```
    return score
```

Скрипт: PathController.cs

```
csharp
```

```
using UnityEngine;
```

```
public class PathController : MonoBehaviour
```

```
{
```

```
    public Transform[] waypoints;
```

```
    public float speed = 5f;
```

```
    private int currentPoint = 0;
```

```
    private float triggerDistance = 2f;
```

```
    public GameObject captureEffectPrefab;
```

```
    void Update()
```

```
    {
```

```
        if (currentPoint >= waypoints.Length) return;
```

```
        Transform target = waypoints[currentPoint];
```

```
        transform.position = Vector3.MoveTowards(transform.position, target.position,  
speed * Time.deltaTime);
```

```
        transform.LookAt(target.position);
```

```
        if (Vector3.Distance(transform.position, target.position) < triggerDistance)
```

```
        {
```

```
            currentPoint++;
```

```
            if (currentPoint == waypoints.Length / 2) // интеллектуальная точка
```

```
            {
```

```
                TriggerShutter();
```

```
            }
```

```

    }
}
public void TriggerShutter()
{
    Debug.Log("Затвор сработал!");
    Instantiate(captureEffectPrefab, transform.position, Quaternion.identity);
}
}

```

Скрипт для кнопки: ShutterButton.cs

```

csharp
using UnityEngine;
public class ShutterButton : MonoBehaviour
{
    public PathController drone;

    public void OnButtonPress()
    {
        drone.TriggerShutter();
    }
}

```