

Балашов К. А.

Научный руководитель: старший преподаватель кафедры ИС Булаев А. В.
*Муромский институт (филиал) федерального государственного бюджетного
образовательного учреждения высшего образования «Владимирский государственный
университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
kirill.balashov12@mail.ru*

Разработка отказоустойчивого кластера на базе K3s с интеграцией Serph в условиях гетерогенной среды

Современные распределённые системы всё чаще строятся по принципам микросервисной архитектуры и использования контейнеризации, однако эксплуатация большого количества серверов оказывается для малых компаний чрезвычайно затратной. Данное исследование посвящено созданию отказоустойчивой инфраструктуры на базе упрощённой версии Kubernetes – K3s, объединённой с распределённым хранилищем данных Serph. Кластер развернут на оборудовании разного типа: двух мини-ПК Asus PN40 на архитектуре x86_64 и одноплатных компьютерах – Raspberry Pi 5, Orange Pi 3 LTS и Raspberry Pi 3, все на архитектуре ARM.

Целью работы стало изучение возможностей построения отказоустойчивой вычислительной инфраструктуры на дешёвом и энергоэффективном оборудовании. Научная ценность проекта заключается в демонстрации применимости кластера на мини- и микро-ПК для решения практических задач при ограниченных ресурсах. Такие решения активно задействуются в распределённых вычислениях, таких как локальная обработка данных в IoT-устройствах, автономной робототехнике или удалённых сенсорных сетях; в этих сферах важны минимальная задержка и работоспособность при низкой пропускной способности соединения с центральным дата-центром. Подобные платформы находят применение в образовательных и научных учреждениях для изучения архитектуры распределённых систем, микросервисов и технологий контейнеризации, а также в малом и среднем бизнесе для разработки частных облаков и домашних лабораторий без крупных вложений.

В сравнении с традиционными серверными ПК кластеры на мини- и микро-ПК демонстрируют значительные преимущества – гораздо более низкую стоимость закупки и эксплуатации, высокую энергоэффективность, компактные габариты, возможность размещения в условиях, не предполагающих наличие серверных шкафов и мощной системы охлаждения. Это делает их отличным решением для тестирования, запуска пилотных проектов и масштабирования при ограниченном бюджете.

Однако реализация поставленной задачи сопряжена со значительными сложностями. Гетерогенность аппаратной базы требует обеспечения совместимости программного обеспечения, корректного распределения нагрузки и оптимизации производительности на узлах с различной вычислительной мощностью. Ограниченные ресурсы, например, объём оперативной памяти, производительность процессора, ёмкость накопителей усложняют настройку оркестратора и распределённого хранилища. Дополнительные трудности возникают при организации стабильного сетевого взаимодействия и мониторинге состояния разнородных узлов. Таким образом, создание отказоустойчивого кластера в подобной среде представляет собой сложную инженерную задачу, решение которой способствует формированию новых принципов к построению надёжных распределённых систем на дешёвом и доступном оборудовании.

K3s – это упрощённая версия Kubernetes, разработанная специально для устройств с ограниченными ресурсами. Она поставляется в виде одного бинарного файла, в то время как полноценный k8s требует тщательной настройки и значительных вычислительных ресурсов. K3s включает все необходимые компоненты: контейнерный рантайм containerd, сетевую подсистему Flannel, DNS-сервер CoreDNS, контроллер входящих подключений Traefik и встроенную базу данных etcd.

В кластере все компоненты K3s распределены между всеми шестью узлами, что обеспечивает отказоустойчивость оркестрации. Мастер узлы K3s размещены на двух мини-ПК

и Raspberry Pi 5, остальные узлы выступают в роли агентов. В отличие от K3s, компоненты Ceph – OSD, MON и MGR – развёрнуты только на мощных устройствах: двух мини-ПК Asus PN40 и Raspberry Pi 5. Более слабые узлы не участвуют в хранении данных и контроле Ceph из-за ограниченности ресурсов, выполняя лишь вычислительные задачи под управлением K3s.

В качестве демонстрационного приложения, наглядно иллюстрирующего работу кластера в условиях высокой нагрузки, используется Gitea – Git-платформа, установленная через официальный Helm-чарт. Приложение функционирует в режиме высокой доступности и имеет следующие встроенные компоненты:

- PostgreSQL HA – распределённая СУБД с автоматической репликацией и переключением при сбоях;

- Valkey – распределённое хранилище кэша, обеспечивающее высокую производительность и отказоустойчивость.

Gitea, PostgreSQL HA и Valkey автоматически распределяются с помощью K3s по всем доступным узлам. PersistentVolumeClaims запрашивают тома у Ceph, который формирует реплицированные тома хранения. Благодаря этому сервис Gitea остаётся полностью доступным даже при одновременном выходе из строя двух-трёх узлов, что позволяет пользователям продолжать работать с репозиториями без перерывов.

Ceph – это масштабируемая и отказоустойчивая распределённая система хранения, объединяющая объектное, блочное и файловое хранилище в единой архитектуре. Для распределения данных и организации репликации используется алгоритм CRUSH – Controlled Replication Under Scalable Hashing, не зависящий от централизованных узлов. Основные компоненты Ceph:

- OSD (Object Storage Daemon) – служба, отвечающая за хранение данных, их копирование, восстановление и балансировку нагрузки. Каждый OSD привязан к отдельному физическому накопителю и обеспечивает целостность данных даже при сбоях отдельных дисков;

- MON (Monitor) – модуль, отслеживающий состояние всех OSD и управляющий составом кластера. Для обеспечения отказоустойчивости необходимо нечётное количество мониторов, минимально – три;

- MGR (Manager) – демон, собирающий статистику, предоставляющий интерфейсы для мониторинга и отвечающий за управление балансировкой и планированием.

Сочетание лёгкого оркестратора K3s и распределённого хранилища Ceph даёт возможность:

- создать инфраструктуру, где вычислительные процессы и хранение данных реализованы на одном и том же оборудовании, что уменьшает расходы на выделенные системы хранения;

- обеспечить динамическое выделение томов приложениям через механизмы PersistentVolumeClaim – пользователь может запросить том определённого объёма, и Ceph автоматически создаст его, обеспечит репликацией и предоставит нужному поду;

- обеспечить отказоустойчивость на уровне кластера – при сбое узла его задачи переносятся на другие узлы, а данные остаются доступными благодаря наличию реплик на других OSD.

В работе реализовано и проведено исследование отказоустойчивого кластера на основе K3s и Ceph, который был развёрнут на гетерогенном оборудовании. Показано, что данная связка позволяет эффективно использовать ресурсы маломощных и разнородных устройств, обеспечивая при этом высокую доступность как вычислительной части, так и подсистемы хранения. Результаты могут быть полезны при построении распределённых вычислений, домашних облачных платформ, а также для обучения принципам работы распределённых систем.

Литература

1. K3s Documentation: официальный сайт. – URL: <https://docs.k3s.io/> (дата обращения: 16.03.2026).

2. Kubernetes Documentation: официальный сайт. – URL: <https://kubernetes.io/docs/home/> (дата обращения: 18.03.2026).
3. Ceph Documentation: официальный сайт. – URL: <https://docs.ceph.com/en/latest/> (дата обращения: 23.04.2026).

Власов Д.С.

Руководитель: к.т.н. доцент кафедры ИС Канунова Е.Е.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: kanunovae@list.ru

Система криптографической защиты данных на основе алгебраических структур

Компьютеры становятся мощнее и быстрее поэтому старые шифры перестают быть эффективными для защиты информационных данных. Модернистские подходы, основанные на проблеме факторизации целых чисел и дискретного логарифмирования, зачастую, оказываются уязвимы перед актуальными алгоритмами дешифрования. Исходя из этого случая, актуальным направлением становится использование некоммутативных и неассоциативных алгебраических структур, обеспечивающих более высокую криптостойкость за счет вычислительной сложности обратных операций.

Представьте, что злоумышленник с усовершенствованными видами атак пытается взломать старый шифр. Очевидно, что в эпоху ИИ и новых способов дешифрования совершить это достаточно просто. Поэтому данная система криптографической защиты крайне необходима и важна.

Теоретическим базисом исследования служат теории абстрактной алгебры, теории сложности вычислений и теории эллиптических кривых (теории эллиптических кривых высших порядков). В отличие от традиционных криптосистем (RSA, ELGamal), где используется коммутативная операция умножения в конечных полях, предлагается применять алгебры с несоциативным умножением.

Как показали эксперименты, создание системы криптографической защиты данных на основе эллиптических кривых открывает новые возможности, позволяя достичь эквивалентного или более высокого уровня безопасности по сравнению с RSA при длине ключа 256 бит вместо 3072 бит [4]. На наш взгляд, эллиптические кривые высших порядков, как алгебраическая структура, предоставляют уникальную возможность для реализации асимметричного шифрования, формирования электронной подписи, что способствует повышению производительности, снижению энергопотребления защищённых систем, а также самого уровня защиты этих систем.

Разработанная система предоставляет широкий инструментарий криптографических функций. Она обеспечивает генерацию ключей на основе базовой точки эллиптической кривой, шифрование данных по схеме ECIES, формирование электронной подписи ECDSA и безопасное согласование ключей по протоколу ECDH. Пользователи системы могут применять ее для защиты каналов связи, шифрования файлов и аутентификации сообщений. Кроме этого, система позволяет выполнять скалярное умножение точки в постоянном времени, что защищает от кибератак по побочным каналам, а также ведет мониторинг создания ключей через механизм обнаружения аномалий.

Литература

1. Глухов, М. М., И. А. Круглов. Элементы теории обыкновенных представлений и характеров конечных групп с приложениями в криптографии: учебное пособие. – Санкт-Петербург: Лань, 2022. – 176 с.

2. Болотов А.А., Гашков С.Б., Фролов А.Б., Часовских А.А. Элементарное введение в эллиптическую криптографию. Алгебраические и алгоритмические основы. – М.: КомКнига, 2006. – 328 с.

3. Соколов А.А. Формирование цифровой подписи на основе эллиптических кривых – Вестник магистратуры. – 2015. – 25-30 с.

4. Обухов В.А. Криптография на основе эллиптических кривых. – электронный научный журнал Ферагнского филиала ТАТУ имени Мухаммада аль-Хоразми. – 2023. 182-188 с.

5. Попова Е.А., Яковлев А. В., Королькова И.К. Применение эллиптических кривых высших порядков для защиты информации. – Информационные технологии в науке, управлении, социальной сфере и медицине. – 2022. 1-10 с.

Гудзь К.С.

Научный руководитель: ст. преподаватель каф. ИС А.В. Булаев
 Муромский институт (филиал) федерального государственного бюджетного
 образовательного учреждения высшего образования «Владимирский государственный
 университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
 602264, г. Муром, Владимирская обл., ул. Орловская, 23
 cfifant120l@gmail.com

Оптимизация хранения компонент в алгоритме построения топологического спектра

В задачах компьютерного зрения и обработки изображений важное место занимает топологический анализ — выделение связных областей (компонент) на бинарных или полутоновых изображениях. Классический алгоритм в общем виде работает следующим образом. Исходная фотография преобразуется в матрицу оттенков серого. Далее производится последовательный перебор ячеек по уровням яркости. Для текущей ячейки проверяется наличие соседей, уже отнесённых к существующим компонентам. Если ячейка не имеет соседей-компонент, создаётся новая компонента. Если соседи есть, ячейка присоединяется к одной из них.

При создании новой компоненты формируется пустая матрица, в которую в координатах, соответствующих текущей ячейке матрицы оттенков, записывается единица. После завершения обработки каждой итерации яркости во всех таблицах компонент значения ненулевых элементов увеличиваются на единицу, что позволяет отследить момент и место создания компоненты во времени.

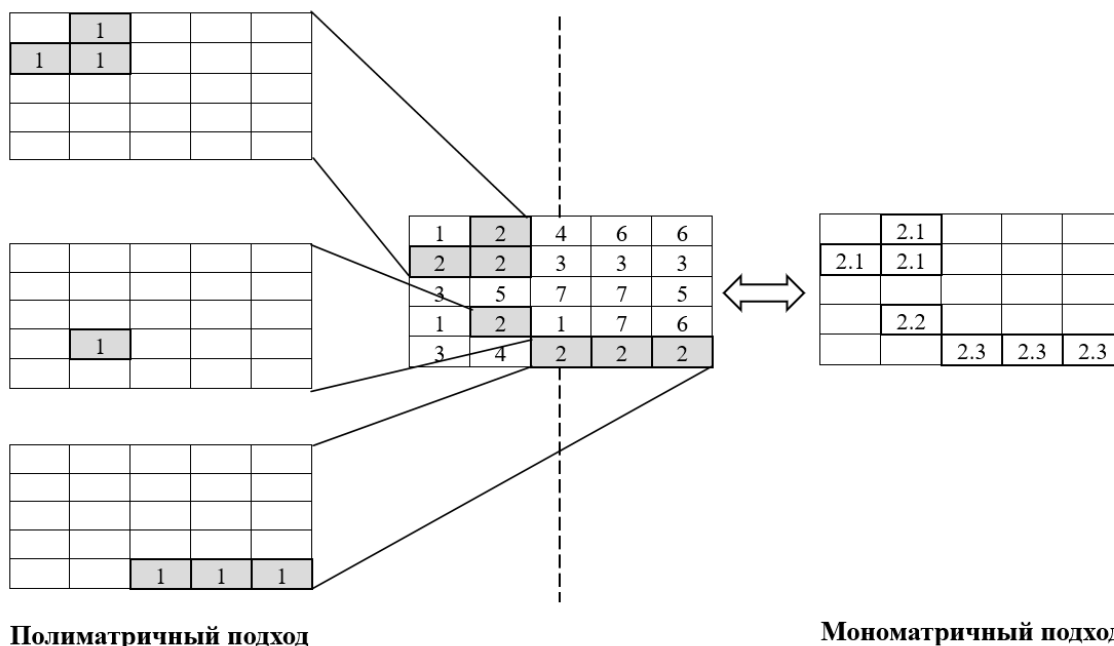


Рис. 1. Хранение матриц компонент при поли- и мономатричном подходах

В результате всего процесса формируется баркод — диаграмма, отображающая для каждой компоненты момент её создания и момент поглощения другой компонентой. У каждого объекта на изображении данный баркод является уникальным, что позволяет использовать его для идентификации и сравнения структур.

Основная проблема классического подхода заключается в том, что при большом количестве создаваемых компонент формируется множество разреженных, полупустых таблиц, что приводит к нерациональному расходу памяти и снижению производительности.

В качестве решения неэффективного использования памяти предлагается упрощённая система хранения данных на основе мономатричного подхода, основанный на использовании единой финальной таблицы компонент. В отличие от классического (полиматричного) подхода

с множеством отдельных матриц, здесь создаётся одна структура данных, размерность которой соответствует размерности исходного изображения.

Каждая ячейка этой таблицы хранит не значение одной компоненты, а кортеж из двух параметров: значение данной ячейки (текущий "возраст" пикселя, увеличивающийся с каждой итерацией) и указатель на компоненту в векторе всех компонент. С помощью указателя на компоненту можно получить всю необходимую информацию о ней: размер компоненты (количество входящих в неё пикселей), максимальное значение компоненты (которое показывает, как давно появилась компонента), а также уникальный идентификационный номер компоненты.

При добавлении новой ячейки к существующей компоненте обновляются её размер и максимальное значение, а при слиянии двух компонент происходит объединение их параметров с сохранением единого идентификатора. Дополнительно в отдельный массив записываются события рождения и поглощения компоненты, что позволяет построить итоговый баркод без хранения множества промежуточных матриц.

Такой подход позволяет отказаться от множества структур и сократить вычислительные затраты за счёт работы с одной таблицей, что особенно важно при анализе высокодетализированных изображений с большим количеством мелких компонент. Финальные результаты при использовании двух подходов – не отличаются.

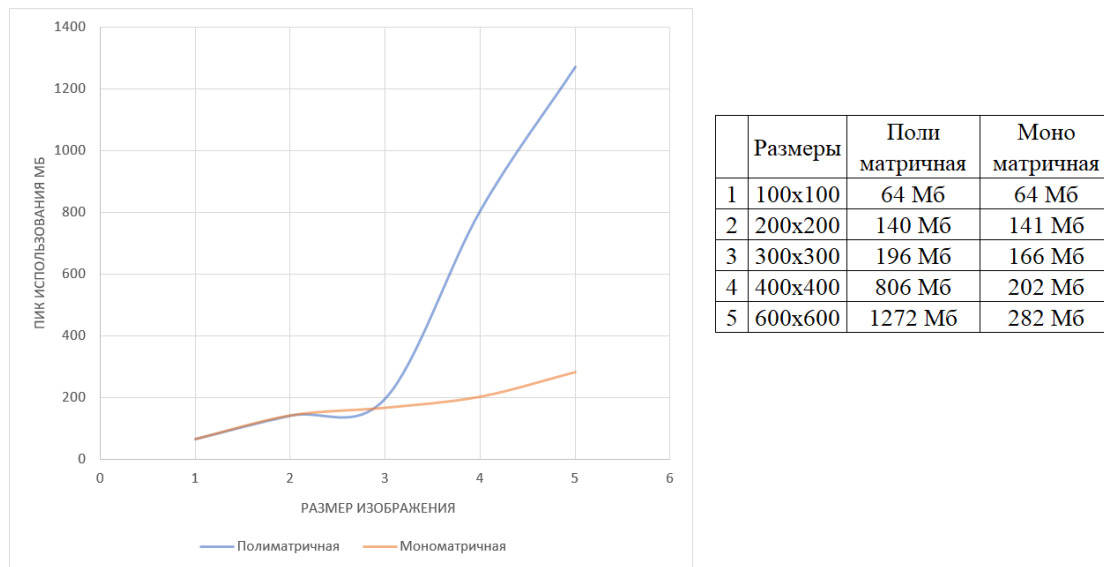


Рис. 2. Оценка потребления памяти при поли- и мономатричном подходах

В докладе будет представлено подробное описание упрощённого алгоритма топологического анализа, продемонстрирована генерация уникальных баркодов для различных объектов и приведены результаты сравнения эффективности предложенного метода с классическим.

Литература

1. Метод разложения изображения по топологическим признакам / С. В. Еремеев, А. В. Абакумов, Д. Е. Андрианов, Д. В. Титов // Компьютерная оптика. – 2022. – Т. 46, № 6. – С. 939-947. – DOI 10.18287/2412-6179-CO-1080. – EDN FPBEUC.

2. Андрианов, Д. Е. Модели и алгоритмы обработки и анализа данных в геоинформационных системах / Д. Е. Андрианов, С. В. Еремеев, А. В. Абакумов. – Москва : Всероссийский научно-исследовательский институт геологических, геофизических и геохимических систем, 2020. – 98 с. – ISBN 978-5-8481-0246-8. – EDN ПХСОР.

3. Метод векторизации спутниковых снимков на основе их разложения по топологическим особенностям / С. В. Еремеев, А. В. Абакумов, Д. Е. Андрианов, Т. А. Ширабакина // Информатика и автоматизация. – 2023. – Т. 22, № 1. – С. 110-145. – DOI 10.15622/ia.22.1.5. – EDN ROYLBV.

Земсков С. А.

Научный руководитель: к.т.н., доцент каф. ИС Щаников С.А.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602263, Муром, Владимирской области, ул. Орловская 23
E-mail: z16124@yandex.ru

Разработка системы визуального трекинга колесного робота на основе компьютерного зрения

Современная робототехника предъявляет высокие требования к точности позиционирования мобильных платформ и безопасности их взаимодействия с окружающей средой. Традиционные методы одометрии подвержены накоплению ошибок, а лазерные дальнометры имеют высокую стоимость. Визуальный трекинг на основе технического зрения является экономичной и информативной альтернативой, позволяющей решать задачи навигации и контроля окружения [1, 2]

Существующие методы компьютерного зрения для трекинга можно разделить на три категории: алгоритмы на основе ключевых точек (ORB, SIFT), методы оптического потока и маркерные подходы. Первые два метода требуют значительных вычислительных ресурсов и чувствительны к условиям съемки, что ограничивает их применение в лабораторных экспериментах, требующих высокой точности при ограниченных вычислительных мощностях [3, 4].

Предлагаемый подход основан на маркерном методе с использованием цветовой сегментации в HSV-пространстве. Этот метод обеспечивает высокую точность определения координат и низкую вычислительную нагрузку за счет использования простых пороговых фильтров и морфологических операций. Ключевыми преимуществами являются стабильность к текстуре фона и возможность работы в реальном времени [3, 4].

Разработанная система имеет модульную архитектуру, включающую блоки захвата видео, предобработки (маскирование ROI), трекинга (поиск контура маркера), анализа безопасности и визуализации. Интерактивный интерфейс позволяет гибко настраивать область анализа и размещать виртуальные препятствия. Для повышения качества отображения траектории применяется алгоритм сглаживания, подавляющий высокочастотный шум детектирования.

Экспериментальные исследования подтвердили эффективность разработанной системы. Точность позиционирования составила 97,5% (относительная ошибка 2,5%) при частоте обработки 30 кадров в секунду. Модуль безопасности продемонстрировал корректное срабатывание трехуровневой системы предупреждений (SAFE, WARNING, DANGER) с временем отклика менее 0,1 секунды. Полученные результаты подтверждают гипотезу о возможности создания эффективной системы трекинга на основе предложенного подхода для контролируемых лабораторных условий.

Литература

- [1] Bradski, G., Kaehler, A. Learning OpenCV: Computer Vision with the OpenCV Library. — O'Reilly Media, 2008.
- [2] Szeliski, R. Computer Vision: Algorithms and Applications. — Springer, 2010.
- [3] Официальная документация библиотеки OpenCV 4.x. Доступ: <https://docs.opencv.org/> (обращение: 10.10.2025).
- [4] Gonzalez, R.C., Woods, R.E. Digital Image Processing. — Pearson, 2018.

Котов Д.С., Марин С.М.

Руководитель: к.т.н. доцент кафедры ИС Канунова Е.Е.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: support@schedule-app.ru

Разработка мобильного приложения с интеллектуальным ассистентом для исследования историко-культурного наследия малых городов России

Малые исторические города России обладают значительным культурным потенциалом, однако их туристическая привлекательность остаётся недостаточно раскрытой. Информация о памятниках архитектуры, выдающихся личностях и исторических событиях, связанных с этими городами, зачастую фрагментарна, рассредоточена по разрозненным источникам и труднодоступна для самостоятельного туриста. Существующие решения - бумажные путеводители, аудиогиды и справочные порталы - не обеспечивают персонализации маршрутов с учётом индивидуальных интересов пользователя. В рамках данной работы разработано мобильное приложение для платформы Android, предназначенное для формирования и прохождения тематических туристических маршрутов по историческим местам малых городов с использованием интеллектуального ассистента.

Приложение представляет собой каталог туристических маршрутов, включающий описания историко-культурных объектов, биографические справки о связанных с ними выдающихся личностях и архивные фотоматериалы. Маршруты визуализируются на интерактивной карте с отображением точек интереса и навигационных указаний. Ключевой функциональной особенностью является встроенный интеллектуальный ассистент, осуществляющий персональный подбор маршрутов на основе предпочтений пользователя: тематических интересов, располагаемого времени и предпочтительного способа перемещения.

Клиентская часть приложения разработана нативно для платформы Android с использованием фреймворка Jetpack Compose и языка программирования Kotlin. Выбор нативной разработки обусловлен необходимостью обеспечения высокой производительности, полноценной интеграции с картографическими сервисами и системными API устройства. Серверная часть реализована на языке Python с применением фреймворка FastAPI и обеспечивает предоставление данных о маршрутах и объектах через REST API. Сетевое взаимодействие на стороне клиента осуществляется с использованием HTTP-клиента Ktor. Интеллектуальный ассистент функционирует на основе интеграции с API большой языковой модели (LLM), которой передаётся контекст доступных маршрутов и пользовательский запрос для формирования персонализированных рекомендаций.

Архитектура проекта построена по принципу клиент-серверного взаимодействия с разделением ответственности. Клиентское приложение реализует паттерн MVVM (Model-View-ViewModel) с использованием компонентов Android Architecture Components, включая ViewModel, StateFlow и Room для локального кэширования данных. Картографический модуль обеспечивает отображение маршрутов и точек интереса на интерактивной карте.

Приложение разрабатывается и тестируется на материале города Муром Владимирской области. В каталог включены тематические маршруты, посвящённые историческим личностям - В.К. Зворыкину, святым Петру и Февронии, былинному Илье Муромцу, - а также архитектурному наследию: торговым рядам, купеческим особнякам и культовым сооружениям. Перспективы развития проекта предусматривают расширение географии на другие малые исторические города.

Крайнов С. А.

Научный руководитель: к.т.н., доцент Еремеев С.В.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых».

602264, Владимирская область, г. Муром, ул. Орловская, д.23

E-mail: an0nim2020@ya.ru

Разработка компактной системы интеллектуального распознавания рукописных выражений для маломощных устройств

Современное развитие интернета вещей и мобильной робототехники требует внедрения алгоритмов машинного зрения непосредственно на конечных устройствах. Главным барьером для обработки рукописной информации на таких платформах является нехватка вычислительных мощностей для запуска глубоких нейронных сетей. В то же время, использование базовых легковесных архитектур приводит к резкому падению качества классификации, особенно при анализе зашумленных документов и визуально похожих символов из разных алфавитов.

Целью данного исследования является создание энергоэффективного алгоритма, способного не просто распознавать рукописный текст, но и логически интерпретировать его (на примере математических уравнений) без использования облачных серверов.

Для решения проблемы предложен гибридный конвейер, сочетающий легковесную нейронную сеть и логический анализатор. Чтобы повысить точность извлечения визуальных признаков без увеличения размера модели, в архитектуру был внедрен модуль пространственно-канального внимания[1]. Этот механизм помогает алгоритму игнорировать фон изображения и сосредотачиваться на уникальных чертах конкретных символов.

Процесс обучения модели базируется на концепции передачи опыта (дистилляции знаний)[2]. Вместо стандартного обучения на сырых данных, наша компактная сеть перенимает обобщенные представления от группы изолированных нейросетей-экспертов, каждая из которых специализируется на узкой языковой области (цифры, латиница, кириллица). Финальным этапом работы системы выступает символьный вычислитель, который преобразует массив предсказанных сетью вероятностей в структурированное абстрактное дерево и осуществляет логический вывод[3].

Тестирование разработанного комплекса на синтетических выборках (свыше 380 тысяч изображений) показало значительный прирост производительности. Предложенный ансамблевый подход с фильтрацией внимания позволил алгоритму достичь рабочей точности в 90% всего за одну эпоху обучения, в то время как стандартная многочитательская дистилляция на тех же данных показала результат менее 50%. При этом время обработки одного символа на стандартном процессоре без использования видеокарты составило около 45 миллисекунд.

Таким образом, интеграция модулей внимания с методами дистилляции открывает путь к созданию автономных систем оптического распознавания, способных быстро и точно считывать математический текст на устройствах с жесткими ресурсными ограничениями.

Литература

1. Woo S., Park J., Lee J. Y., Kweon I. S. CBAM: Convolutional Block Attention Module // Proceedings of the European Conference on Computer Vision (ECCV). 2018. P. 3–19.
2. Hinton G., Vinyals O., Dean J. Distilling the Knowledge in a Neural Network // NIPS Deep Learning Workshop. 2015.
3. Garcez A. D., Lamb L. C. Neurosymbolic AI: The 3rd Wave // Artificial Intelligence Review. 2023. Vol. 56. P. 12387–12405.

Логашов А.А.

Руководитель: к.т.н. доцент кафедры ИС Канунова Е.Е.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: kanunovae@list.ru

Разработка мобильного приложения для выполнения лабораторных и практических работ по информатике

В современном мире технологии играют ключевую роль в оптимизации различных процессов, в том числе в сфере образования. Одной из приоритетных задач для учебных заведений остаётся упрощение учебного процесса. В рамках данного проекта разработана программа, предназначенная для оптимизации выполнения практических и лабораторных работ по информатике.

Лабораторные работы направлены на освоение методов работы с информацией, изучение систем счисления и приобретение навыков эксплуатации прикладного программного обеспечения.

Практические работы направлены на освоение навыков работы в интернет-среде, изучение основ алгебры логики и системологии, а также на знакомство со средой разработки Qt Creator.

Оптимизация учебного процесса достигается за счет прямой интеграции заданий в приложение, что исключает необходимость перехода на Информационно-образовательный портал (ИОП). Наличие функции мгновенного ввода и автоматической проверки ответов существенно снижает нагрузку на преподавателя, освобождая его от ручной сверки и позволяя анализировать уже готовые результаты.

Приложение разработано с использованием объектно-ориентированного подхода, что облегчает разработку проекта и его поддержку в будущем. Функциональная часть реализована на языке Kotlin. Интерфейс написан с помощью фреймворка Android XML-layout. В качестве среды разработки использовалась Google Android Studio.

Программа встречает пользователя главным меню, включающим в себя список доступных работ и полем для ввода индивидуального Мп-кода.

Для студента реализована возможность просмотра заданий работ, а также ввода и проверки ответов на них.

Интерфейс приложения ориентирован на пользователя, обладает интуитивно понятным функционалом. Тем самым облегчается использование программы.

В будущем планируется расширение списка доступных работ, внедрение поддержки баз данных, создание личных кабинетов для учеников и разработка функционала для преподавателя.

Литература

1. Официальная документация Kotlin. [Электронный ресурс]. URL: <https://kotlinlang.org/docs> (дата обращения: 15.01.2026).
2. Официальная документация Android Studio. [Электронный ресурс]. URL: <https://developer.android.com/develop?hl=ru> (дата обращения: 15.01.2026)

Ногтев С.А., Белогузов А.С., Мурындин Д.С.
Руководитель: к.т.н. доцент кафедры ИС Канунова Е.Е.
*Муромский институт (филиал) федерального государственного бюджетного
образовательного учреждения высшего образования «Владимирский государственный
университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: stepan.nogtev@mail.ru*

Гроза 1812 года: военно-историческая стратегия

В современном образовательном процессе одной из ключевых проблем остаётся недостаток наглядности при изучении крупных исторических событий, в частности Отечественной войны 1812 года. Статичные карты в атласах не позволяют в полной мере оценить масштаб Бородинского сражения или логистику отступления русской армии. Кроме того, популярные зарубежные игры часто искажают роль России в войне 1812 года, а традиционное обучение даёт знание дат без понимания причинно-следственных связей. В связи с этим актуальной становится разработка отечественного игрового продукта с достоверными историческими данными.

Целью проекта является создание компьютерной игры-симулятора, совмещающей управление армией, стратегическое мышление и интерактивную энциклопедию с историческим контекстом. Конечный продукт - компьютерный образовательный симулятор «Гроза 1812».

Разрабатываемая игра «Гроза 1812» позволит преодолеть недостаток наглядности, обеспечить историческую достоверность и внедрить метод «реконструкции событий» в образовательный процесс, что делает её востребованным инструментом геймификации истории в школах и вузах.

Литература

3. 1. Документация Unity Asset Store [Электронный ресурс] // URL:
<https://assetstore.unity.com> (дата обращения: 04.04.2026)

Осинцев Д.Н.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
osincevd@bk.ru

Эффективная 3D-реконструкция в реальном времени с использованием геометрически-осведомленных гауссовых сурфелов (EGG-Fusion)

Развитие систем смешанной реальности, робототехники и автономного транспорта делает задачу высокоточной 3D-реконструкции в реальном времени критически важной. Современные differentiable-rendering SLAM-подходы на базе NeRF и 3D Gaussian Splatting обеспечивают высокое качество визуализации, но в практических RGB-D сценариях часто уступают по геометрической точности и устойчивости.

Ключевые проблемы существующих GS-методов связаны с шумом потребительских depth-сенсоров, геометрической неоднозначностью примитивов и чувствительностью трекинга к быстрому движению камеры. Это приводит к деградации точности поверхностей, локальным артефактам и риску потери трека в сложных условиях съемки.

Целью работы является разработка и исследование подхода EGG-Fusion, объединяющего представление сцены 2D Gaussian surfels, геометрически-осведомленную инициализацию примитивов, слияние сурфелов на основе информационного фильтра и схему sparse-to-dense трекинга камеры для получения точной и вычислительно эффективной онлайн-реконструкции.

В рамках исследования решаются следующие задачи:

1. формализовать и реализовать представление сцены на основе 2D Gaussian surfels и дифференцируемого рендеринга цвета, глубины и нормалей;
2. разработать геометрически-осведомленную инициализацию сурфелов (выбор по низкой непрозрачности и положительному depth-disparity) с адаптивным масштабом, зависящим от глубины и параметров камеры;
3. реализовать слияние сурфелов с информационным фильтром и рекурсивным обновлением состояния (позиция, нормаль) и матрицы неопределенности для подавления шумов датчика;

На рис. 1 показано место EGG-Fusion в прикладном конвейере RGB-D реконструкции: входные кадры поступают в модуль sparse-to-dense трекинга и в модуль построения карты на Gaussian surfels, где выполняются инициализация, фильтрационное слияние и дифференцируемая оптимизация.

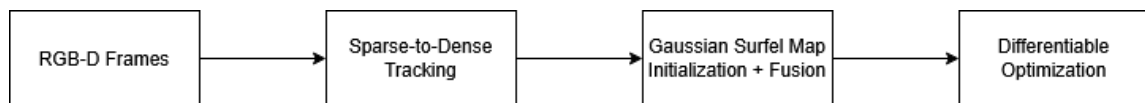


Рис. 1. Структура конвейера EGG-Fusion для онлайн 3D-реконструкции

Литература

1. Pan X., Li Z., Ye Z., Zhai H., Zhang G. EGG-Fusion: Efficient 3D Reconstruction with Geometry-aware Gaussian Surfel on the Fly. arXiv:2512.01296, 2025.
2. Kerbl B., Kopanas G., Leimkühler T., Drettakis G. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. ACM TOG, 2023.
3. Newcombe R. A., Izadi S., et al. KinectFusion: Real-Time Dense Surface Mapping and Tracking. ISMAR, 2011.
4. Whelan T., Salas-Moreno R., Glocker B., Davison A., Leutenegger S. ElasticFusion: Dense SLAM Without A Pose Graph. RSS, 2015.
5. Keetha N., Karhade J., et al. SplatTAM: Splat, Track & Map 3D Gaussians for Dense RGB-D SLAM. CVPR, 2024.

Парамонов Н.С.
Научный руководитель к.т.н., доцент С.В. Еремеев
*Муромский институт (филиал) федерального государственного бюджетного
образовательного учреждения высшего образования «Владимирский государственный
университет имени Александра Григорьевича и Николая Григорьевича Столетовых»*
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
E-mail: paramonov678@gmail.com

Автономная интеграция IoT-устройств в корпоративные системы с безопасной аутентификацией

Российский рынок промышленного IoT демонстрирует устойчивый рост: по оценкам TAdviser, в 2024 году объём сегмента превысил 150 млрд рублей, а ежегодный прирост до 2026 года прогнозируется на уровне 12 - 18 % на фоне курса на импортозамещение и реализации национального проекта «Экономика данных» [1]. Вместе с тем преобладающая ориентация на облачные платформы порождает системные риски: утечки данных, коммуникационные задержки свыше 150 - 250 мс, а также зависимость от зарубежных провайдеров в условиях санкционных ограничений и требований локализации [2].

В ответ на эти вызовы предлагается автономная система, в которой регистрация, аутентификация и обмен данными реализованы исключительно в изолированной корпоративной сети. Архитектура включает: офлайн-регистрацию устройств с верификацией физического владения через Serial и HMAC-проверку микроконтроллеров; постквантовое первоначальное согласование ключей на основе протокола Noise_XX + ML-KEM-1024 (Kyber) с аппаратной привязкой через механизм efuse; непрерывную ротацию сеансовых ключей с интервалом 300 секунд или 300 пакетов для обеспечения прямой секретности (PFS) и защиты после компрометации (PCS); периодический аудит целостности прошивки.

Научная новизна состоит в оригинальной комбинации постквантовой стойкости, аппаратной привязки идентификатора и непрерывной ротации ключей, реализованной на ресурсно-ограниченных микроконтроллерах (Seeed Studio XIAO ESP32-C6 и ESP32-S3, 4-8 МБ Flash). Патентный анализ (RU2695487C1, RU2767713C1, RU192192U1, US20170279620A1, US10171462) подтверждает отсутствие прямых аналогов: существующие решения либо используют облачную инфраструктуру, либо не обеспечивают достаточного уровня защиты.

Для преодоления известных ограничений постквантовой криптографии - увеличенных размеров ключей и шифротекстов, высоких вычислительных затрат, уязвимостей к атакам по сторонним каналам - применяется авторский алгоритм ЛАНШИТО (Легковесное Автономное Непрерывное Шифрование с Использованием Токенов Обновления). Алгоритм объединяет Noise_XX, ML-KEM и SLH-DSA и при ожидаемых сопоставимых показателях безопасности нацелен на снижение энергопотребления относительно чистого Kyber.

Практическая значимость разработки ориентирована на промышленность, энергетику и объекты критической информационной инфраструктуры (КИИ), где определяющими требованиями являются суверенитет данных и минимальные задержки.

Литература

1. TAdviser. Интернет вещей, IoT, M2M (рынок России). URL: [https://www.tadviser.ru/index.php/Статья:Интернет_вещей,_IoT_\(рынок_России\)](https://www.tadviser.ru/index.php/Статья:Интернет_вещей,_IoT_(рынок_России)) (дата обращения: 27.12.2025).
2. Palo Alto Networks. How to Secure IoT Devices in the Enterprise. URL: <https://www.paloaltonetworks.com/cyberpedia/how-to-secure-iot-devices-in-the-enterprise> (дата обращения: 27.12.2025).

Рассказов А.Р.

Научный руководитель: ст. преподаватель каф. ИС Булаев А.В.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
 г. Муром, Владимирская обл., ул. Орловская, 23
 aleksandr.rasskazov.2019@gmail.com

Информационная система для управления учебной нагрузкой преподавателей вуза

В большинстве высших учебных заведений процесс планирования и распределения учебной нагрузки преподавательского состава выполняется вручную с использованием электронных таблиц. Исходными данными служит файл «Общая нагрузка кафедры» в формате Excel или Calc, содержащий до двух-трёх сотен строк с информацией о дисциплинах, группах, курсах, количестве студентов и часах по видам работ: лекции, практические и лабораторные занятия, консультации, приём экзаменов и зачётов, руководство курсовыми проектами и практиками. Основная проблема заключается в том, что каждый преподаватель вынужден вручную находить в этом общем файле строки с собственной фамилией, копировать их в свой шаблон индивидуального плана, а затем суммировать часы по каждому виду работ. При внесении изменений в общую нагрузку в середине учебного года, всю эту процедуру приходится повторять заново. Такой подход приводит к значительным временным затратам, арифметическим ошибкам, несогласованности данных между разными файлами и сложностям с контролем допустимых норм часов для разных категорий персонала — штатных преподавателей, совместителей и почасовиков.

Для формализации процесса расчёта предлагается следующая математическая модель (1). Общая учебная нагрузка преподавателя по всем закреплённым за ним дисциплинам представляет собой сумму аудиторной и внеаудиторной составляющих:

$$W_{\text{общ}} = \sum_{i=1}^n (A_i + B_i) \quad (1)$$

где A_i — аудиторная нагрузка по i -й дисциплине, включающая часы лекций, практических и лабораторных занятий; B_i — внеаудиторная нагрузка по i -й дисциплине, включающая консультации, приём экзаменов и зачётов, проверку контрольных работ, руководство курсовыми проектами и выпускными квалификационными работами, а также руководство практикой.

При этом для разных категорий преподавателей применяются корректирующие коэффициенты. Штатный сотрудник на полную ставку может иметь нагрузку в пределах установленного максимума, например, 900 часов в год, и вся его нагрузка целиком входит в основной индивидуальный план. Совместитель ограничен половиной от этой нормы, причём часы по отдельным дисциплинам могут учитываться с понижающим коэффициентом в зависимости от формы занятости. Почасовику оплачиваются только фактически проведённые часы, и в его индивидуальный план попадают исключительно строки, помеченные как почасовая оплата.

Для реализации предлагаемой системы был проведён анализ существующих подходов к хранению данных. Использование электронных таблиц не обеспечивает целостности и единого хранения информации — каждый преподаватель ведёт свой файл, а при изменениях возникает необходимость в полном ручном пересчёте. Локальные базы данных типа SQLite решают проблему структурированности, но не позволяют организовать одновременную работу нескольких пользователей и централизованный доступ к актуальной версии нагрузки. Профессиональные клиент-серверные СУБД, такие как PostgreSQL или MySQL, требуют выделенного сервера, его установки, настройки резервного копирования и постоянного администрирования, что создаёт дополнительную нагрузку на IT-отдел вуза. Поэтому в качестве оптимального решения выбрана облачная реляционная база данных Supabase, работающая на основе PostgreSQL. Supabase предоставляет готовое централизованное хранилище с автоматическим резервированием, встроенным REST API и возможностью

синхронизации данных между несколькими десктоп-приложениями в реальном времени. Бесплатный тарифный план покрывает потребности кафедры с числом преподавателей до 30–50 человек и объёмом хранимой информации в несколько сотен мегабайт, при этом разработчику не требуется администрировать сервер или настраивать безопасность подключения.

Система реализуется как desktop-приложение, а не как веб-сервис. Это решение обусловлено тем, что десктоп-версия представляет собой единственный исполняемый файл или установочный пакет, который сотрудник вуза может установить на свой рабочий компьютер за несколько минут без участия системного администратора. Для веб-приложения потребовалось бы разворачивать серверную инфраструктуру, настраивать веб-сервер, обеспечивать HTTPS-соединение, управлять сессиями пользователей и решать вопросы сетевой доступности, что создаёт дополнительную нагрузку на IT-отдел института и требует квалифицированной поддержки на протяжении всего времени эксплуатации. Таким образом, выбор desktop-архитектуры обусловлен практичностью, автономностью и минимальными требованиями к инфраструктуре вуза.

Предлагаемая система разрабатывается на языке Java с использованием фреймворка JavaFX для современного графического интерфейса, что обеспечивает кроссплатформенность и работу на любых операционных системах. Процесс работы с системой выглядит следующим образом. Сотрудник учебного отдела выгружает из «Скалы» файл общей нагрузки кафедры, после чего загружает этот файл в разрабатываемую систему. Система считывает все строки с дисциплинами, группами, часами и закреплёнными преподавателями. Далее система на основе математической модели (1) автоматически рассчитывает итоговую нагрузку для каждого сотрудника. Пользователь может настроить правила распределения, указав, например, что лекции и практики должны идти в основную нагрузку, а консультации и проверка контрольных работ — в почасовую оплату.

Система автоматически проверяет, не превышает ли суммарная нагрузка штатного преподавателя установленный максимум, а для совместителей — ограничение в 0,5 ставки. При превышении нормы система выдаёт предупреждение и не позволяет сохранить некорректный индивидуальный план.

После автоматического расчёта формируются черновики индивидуальных планов, которые допускают ручную корректировку: можно перенести часы из одного раздела в другой, добавить или удалить дисциплину, причём система пересчитывает итоговые суммы в реальном времени. При изменении исходной нагрузки, сотрудник загружает из «Скалы» обновлённый файл, система автоматически находит изменившиеся строки и пересчитывает только затронутые индивидуальные планы без потери ранее внесённых ручных правок. Финальным этапом является экспорт готовых документов: индивидуальные планы преподавателей выгружаются в формате MS Word или LibreOffice Writer для печати и утверждения, а сводные таблицы с нагрузкой — в формате MS Excel или LibreOffice Calc для дальнейшей аналитики.

Внедрение разработанной системы позволит сократить время, затрачиваемое преподавателями и сотрудниками учебных отделов на формирование и актуализацию индивидуальных планов, в несколько раз. Будет обеспечено единое централизованное хранение данных, исключающее их потерю и несогласованность. В отличие от громоздких и дорогостоящих систем типа «ИС:Университет ПРОФ», предлагаемое решение является лёгким, бесплатным и заточенным исключительно под задачу формирования индивидуальных планов преподавателей на основе выгрузки из «Скалы». Ключевым результатом станет снижение количества ошибок при расчетах, автоматический контроль соблюдения нормативов нагрузки для всех категорий преподавателей и, как следствие, повышение прозрачности и эффективности процесса планирования учебной работы на кафедре.

Сеничкин Д.О.

Научный руководитель: к.т.н. Белякова А.С.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых» 602264 г.

Муром, Владимирской обл., ул. Орловская, д. 23

dsenichkin@yandex.ru

Подход к распознаванию объектов круглого леса на изображениях

Древесина, особенно круглый лес (спиленные стволы без веток и верхушки), является ключевым природным ресурсом для строительства, мебельной промышленности и энергетики. При транспортировке автомобилями с открытым кузовом критически важным становится точный контроль объёма груза: от этого зависят расчёты стоимости, соблюдение транспортных ограничений и предотвращение перегруза, способного привести к аварии. Традиционные ручные измерения трудоёмки, субъективны и сопровождаются ошибками. Автоматизация на основе компьютерного зрения с использованием обычных фотографий с камер наблюдения или мобильных устройств позволяет сократить время оценки и повысить объективность. В связи с развитием глубоких нейронных сетей задача локализации и распознавания объектов (в частности, брёвен круглой формы) становится всё более актуальной.

В работе изучены возможности нейросетевых моделей для распознавания объектов определённой формы (круглый лес) на фотографиях. В перспективе это позволит автоматически рассчитывать объём древесины по формулам через диаметр бревна в пикселях, масштаб (м/пиксель) и длину бревна. Для достижения цели необходимо рассмотреть альтернативные методы распознавания, обучить модель на малом датасете и проанализировать результаты.

Было сформировано 45 размеченных изображений и три дополнительных без разметки для проверки. Разметка выполнена в инструменте CVAT (Computer Vision Annotation Tool) вручную по наиболее крупным и заметным торцам брёвен. Несмотря на малый объём, такой подход допустим для исследовательских работ с оценкой применимости метода [1].

В качестве архитектуры модели выбрана свёрточная нейронная сеть YOLO (однопроходное детектирование), которая позволяет обнаруживать объекты быстро и эффективно, пригодна для реального времени и демонстрирует приемлемую работоспособность даже при ограниченном объёме обучающих данных. Использована лёгкая версия YOLOv8-seg, библиотека Ultralytics. Другие методы: haag-каскады (быстры, но плохо работают на сложных объектах), LBP (ограничены в сложных сценах), HOG (уступает глубоким сетям по точности), MLP и simpleNet (требуют ручной сегментации, плохо справляются с локализацией) значительно уступают YOLO по качеству и устойчивости [2].

Модель обучена на 42 изображениях, протестирована на 4 контрольных кадрах, не участвовавших в обучении. Для оценки качества детектирования учитывалась визуальная точность позиционирования ограничивающих прямоугольников.

В результате исследования было получено:

- Модель успешно обнаруживает круги брёвен даже на сложных сценах с неоднородным фоном.

- Зафиксировано единичное ложное срабатывание (на водяном знаке), что указывает на чувствительность к контрастным элементам изображения.

- Ограничивающие прямоугольники характеризуются высокой точностью позиционирования, что потенциально позволяет использовать координаты для последующего вычисления геометрических параметров (например, диаметра бревна).

Исследование показало, что архитектура YOLO демонстрирует высокий потенциал для задачи детектирования брёвен на изображениях. Работа носила экспериментальный характер и подтвердила целесообразность дальнейших исследований в этом направлении.

Литература

1. Шелиски Р. Computer Vision: Algorithms and Applications. – London: Springer, 2010.
2. Принс С. Машинное обучение. От основ до продвинутых моделей. – СПб.: Питер, 2025.

Филинов А.Е.

Научный руководитель: ст. преподаватель каф. ИС Булаев А.В.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
г. Муром, Владимирская обл., ул. Орловская, 23
filinov.andrey22@yandex.ru

Информационная система управления городским транспортом г. Муром

Управление маршрутной сетью городского пассажирского транспорта в малых и средних городах России по-прежнему строится через СМИ. В г. Муром взаимодействие между коммерческими перевозчиками и органами местного самоуправления осуществляется через бумажные заявки и совещания: от подачи заявки на новый маршрут до его выхода на линию проходят недели. Пассажиры лишены единого источника актуального расписания, диспетчеры не имеют инструмента для оперативного мониторинга движения транспортных средств. Отсутствие автоматизации обнаруживает пять взаимосвязанных проблем: длительные сроки согласования маршрутов, разрозненность данных о расписании, невозможность мониторинга транспорта в реальном времени, ручная коррекция изменений и отсутствие у пассажиров доступа к актуальной информации [1].

Анализ существующих решений показывает, что они делятся на два непересекающихся класса. Информационно-справочные системы (2ГИС, Московское транспортное приложение) решают задачу отображения данных для пассажиров, однако не предоставляют инструментов для перевозчиков и администрации: подача заявок, согласование маршрутов и управление расписанием остаются вне их функционала. Диспетчерские платформы (CityBus, государственная АСУ ГППТ Санкт-Петербурга, платформа «Диспетчеризация ЭРА» на базе ГЛОНАСС) обеспечивают мониторинг транспорта, но также не автоматизируют документооборот между перевозчиком и администрацией [2, 3]. Кроме того, государственные системы не применимы в малых городах без значительных капитальных вложений.

Таблица 1 — Сравнение систем-аналогов по ключевым функциям

Функция	2ГИС	CityBus	АСУ ГППТ	Пл. ЭРА	Разраб. система
Карта маршрутов и остановок	+	+	+	+	+
Электронная подача заявки	–	–	–	–	+
Согласование с администрацией	–	–	–	–	+
Управление расписанием	–	–	+	+	+
Проверка конфликтов расписания	–	–	–	–	+
GPS-мониторинг транспорта	–	+	+	+	+
Ролевая модель доступа	–	–	частично	–	+

Таким образом, ни одно из рассмотренных решений не закрывает нишу, в которой нуждается малый город: единый веб-инструмент для автоматизации полного цикла — от подачи заявки перевозчиком до информирования пассажира.

Предлагается разработка информационной системы управления городским транспортом г. Муром — веб-приложения с клиент-серверной архитектурой, реализованного на платформе Java 21 + Spring Boot 3.4, PostgreSQL и Яндекс.Картах API. Система включает четыре ролевых модуля: для пассажиров, перевозчиков, администраторов и диспетчеров.

Для перевозчиков реализован механизм электронной подачи заявок на маршруты: выбор остановок из базы данных, задание порядка следования, ввод расписания с автоматической проверкой временных конфликтов — система сигнализирует об интервале менее 5 минут между маршрутами на одной остановке. Для администраторов предусмотрена панель рассмотрения заявок с возможностью одобрения (включая корректировку расписания) или отклонения. Пассажирам доступна интерактивная карта на основе Яндекс.Карт с фильтрацией маршрутов и просмотром расписания. Диспетчеры получают отображение текущих GPS-координат транспортных средств в реальном времени через WebSocket-соединение [4].

Безопасность системы обеспечивается сессионной аутентификацией Spring Security с BCrypt-хэшированием паролей и верификацией e-mail при регистрации через Яндекс SMTP. Ролевая модель (ROLE_ADMIN, ROLE_CARRIER, ROLE_USER, ROLE_DISPATCHER) реализована на уровне каждого эндпоинта REST API. Схема базы данных включает десять таблиц с ссылочной целостностью; все многошаговые операции выполняются транзакционно.

Ожидается, что внедрение разработанной системы позволит сократить время согласования маршрутов с нескольких недель до нескольких дней за счёт перевода документооборота в электронный вид. Единая база расписания с контролем конфликтов снизит вероятность наложения маршрутов на одних остановках. Пассажиры г. Муром получают веб-интерфейс с актуальной информацией о маршрутах и расписании без необходимости устанавливать мобильное приложение. Диспетчеры смогут контролировать движение транспорта в реальном времени через единый интерфейс.

В числе планируемых к реализации функций — механизм динамического перестроения маршрутов. При переводе маршрута в статус «на ремонте» или «временно закрыт» система автоматически предложит пассажирам альтернативные маршруты через ближайшие к закрытым остановки и уведомит об изменении через веб-интерфейс. Диспетчер и администратор получают инструмент для быстрого задания временных обходных маршрутов без полного цикла согласования. Наряду с этим планируется интеграция с push-уведомлениями об изменениях расписания, аналитический модуль (загруженность остановок, интервальность движения) и мобильное приложение для диспетчеров. Архитектура системы допускает тиражирование на другие малые города России без изменения серверного ядра.

Литература

1. Коноплёва И.А. Информационные технологии: учебное пособие. — М.: Проспект, 2020. — 328 с.
2. Smart City Lab. Рейтинг транспортных приложений мегаполисов России. — М.: 2024. — 45 с.
3. Министерство транспорта РФ. Концепция развития ГИС на транспорте. — М.: 2023. — 78 с.
4. Craig Walls. Spring in Action, 6th Edition. — Manning Publications, 2022. — 520 p.

Харьков Н. В.
ст. преподаватель каф. ИС Булаев А.В.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, Россия, Владимирская область, г. Муром, ул. Орловская, 23
E-mail: *ivan.ivanovich23451@mail.ru*

Использование топологического баркода ROI для сопоставления с базой ориентиров в задаче визуальной навигации БПЛА

При работе беспилотного летательного аппарата (БПЛА) над местностью сигналы глобальной навигационной спутниковой системы (ГНСС) могут быть нестабильны или недоступны. Поэтому для резервирования навигации применяются методы машинного зрения, позволяющие сопоставлять текущее изображение сцены с заранее подготовленной базой ориентиров и уточнять положение аппарата [1].

В задачах визуальной навигации используются несколько групп технологий обработки данных: методы по визуальным меткам, методы по ключевым точкам и дескрипторам, визуальная одометрия и SLAM (Simultaneous Localization and Mapping - одновременная локализация и построение карты), построение мозаики местности, а также VPR (Visual Place Recognition - визуальное распознавание места) [1-3]. При этом современные обзоры показывают, что в прикладных системах все заметнее доминируют методы глубокого обучения, однако для малогабаритных БПЛА их применение ограничено вычислительными ресурсами и энергопотреблением [1-3].

Публикации по бортовому компьютерному зрению подтверждают компромисс между точностью и скоростью: при переносе на одноплатные вычислители точность детектирования часто сохраняется только ценой снижения частоты обработки кадров, а облегченные версии моделей работают быстрее, но уступают по качеству. Так, для onboard-сценариев на edge-устройствах исследователи фиксируют необходимость выбирать более компактные решения и упрощенные вычислительные схемы [4, 5].

В работе предлагается использовать топологический баркод ROI (Region of Interest - область интереса) как дескриптор ориентира. Под ROI понимается локальный фрагмент кадра или мозаики, содержащий устойчивый и информативный элемент сцены. В прототипе используется окно фиксированного размера 128×128 пикселей, привязанное к кандидату в ориентиры после грубого геометрического выравнивания по мозаике. Баркод формируется не по одному бинарному изображению, а по семейству бинарных представлений ROI, полученных при последовательном изменении порога яркости; далее анализируются появление, расширение и слияние компонент связности [6-8].

Предварительная обработка строится по принципу минимально необходимого контура. В базовом режиме выполняются нормализация яркости и перевод изображения в градации серого, а бинаризация используется как часть многопорогового анализа ROI. Морфологическая фильтрация, утоншение и скелетизация рассматриваются как дополнительные операции и подключаются только для тех ROI, где базовый режим не формирует устойчивую структуру. Такой подход снижает вычислительную нагрузку и делает обработку ближе к режиму реального времени [1, 5].

Идея баркод-дескриптора опирается на русскоязычные работы С. В. Еремеева, А. В. Абакумова, Д. Е. Андрианова и соавторов по топологическому анализу изображений и спутниковых снимков [6-8]. В этих исследованиях показано, что топологические признаки могут уступать нейросетевым моделям по базовой точности, но демонстрируют более устойчивое поведение при искажениях изображения. Это делает их перспективными для учебного прототипа навигационной системы, где важны интерпретируемость, компактность и устойчивость к изменению условий съемки [7, 8].

В предлагаемой системе баркод используется для сопоставления, текущего ROI с базой ориентиров. Для каждого эталонного ориентира сохраняются его ROI, координаты узла на мозаике и топологическое описание. Если степень сходства превышает заданный порог,

ориентир считается распознанным. Такое распознавание используется не вместо визуальной одометрии, а как источник абсолютной поправки, уменьшающей накопленный дрейф при последовательной обработке видеок кадров [3, 6, 9].

Базу ориентиров в прототипе предлагается реализовать на основе системы управления базами данных (СУБД) РЕД База Данных [10]. В ней предполагается хранить идентификатор ориентира, баркод-дескриптор, координаты узла на мозаике, уровень достоверности совпадения, временную метку наблюдения и служебные показатели качества.

Укрупненная структура системы включает видеовход, выделение ROI, построение мозаики местности, стартовую привязку по визуальному маркеру или ориентиру, извлечение баркод-дескриптора, базу ориентиров, модуль сопоставления и модуль коррекции маршрута. Результатом работы должен стать программный прототип, позволяющий оценить скорость вычисления баркод-дескриптора, качество распознавания ориентиров и эффект коррекции маршрута по графику ошибки до и после коррекции [1, 4, 5, 9, 10].

Принятые сокращения: БПЛА - беспилотный летательный аппарат; ГНСС - глобальная навигационная спутниковая система; ROI - область интереса; SLAM - одновременная локализация и построение карты; VPR - визуальное распознавание места; СУБД - система управления базами данных.

Литература

1. Зикратов И. А., Беляев П. Ю., Неверов Е. А. Алгоритм ориентирования на местности беспилотных летательных аппаратов с использованием машинного зрения // Научно-технический вестник информационных технологий, механики и оптики. 2024. Т. 24. № 5. С. 726-737.
2. Каткури А. В. Р., Мадан Х., Хатри Н. и др. Автономная навигация БПЛА с использованием фреймворков компьютерного зрения на основе глубокого обучения: систематический обзор // Array. 2024. Vol. 23. 100361.
3. Москаленко И., Корнилова А., Феррер Г. Визуальное распознавание места для аэроизображений: обзор // Robotics and Autonomous Systems. 2025. Vol. 183. 104837.
4. Ма М.-Й., Шэнь С.-Э., Хуан Ю.-Ц. Повышение качества визуального распознавания посадки БПЛА с использованием детекции объектов YOLO на борту edge-вычислителя // Sensors. 2023. 23(21):8999.
5. Росенде С. Б., Гислер С., Фернандес-Андрес Х. и др. Реализация системы технического зрения с edge-вычислениями на одноплатных компьютерах, встроенных в БПЛА, для интеллектуального управления трафиком // Drones. 2023. 7(11):682.
6. Еремеев С. В., Абакумов А. В., Андрианов Д. Е., Титов Д. В. Метод разложения изображения по топологическим признакам // Компьютерная оптика. 2022. Т. 46. № 6. С. 939-947.
7. Еремеев С. В., Абакумов А. В. Классификация объектов на изображениях с учетом искажений на основе двухэтапного топологического анализа // Научно-технический вестник информационных технологий, механики и оптики. 2022. Т. 22. № 1. С. 82-92.
8. Еремеев С. В., Абакумов А. В., Андрианов Д. Е., Ширабакина Т. А. Метод векторизации спутниковых снимков на основе их разложения по топологическим особенностям // Информатика и автоматизация. 2023. Т. 22. № 1. С. 110-145.
9. Эль-Хадж Л., Корани У., Эль-Хури П., Кайсси А. Интегрированный алгоритм визуальной навигации для городской аэромобильности // Big Data Research. 2024. Vol. 36. 100447.
10. РЕД База Данных // РЕД СОФТ. Официальный сайт.

Шамили́н Д.Д.

Научный руководитель: ст. преподаватель каф. ИС А.В. Булаев
*Муромский институт (филиал) федерального государственного бюджетного
образовательного учреждения высшего образования «Владимирский государственный
университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
danuul.shamilin@yandex.ru*

Алгоритм яркостно-метрического разложения изображения по топологическим признакам

Количество цифровых изображений постоянно растёт, и их нужно быстро обрабатывать. Классические методы компьютерного зрения часто дают сбой: они чувствительны к шумам, теням и изменению освещения. Всё потому, что они опираются только на яркость отдельных пикселей. А яркость на снимках легко искажается из-за бликов, теней, что снижает точность распознавания объектов.

В качестве альтернативы классическим методам выделяют яркостный подход к топологическому разложению изображений [1]. Он работает так: изображение раскладывается на специальные матрицы, каждая из которых соответствует отдельному объекту или связной области. При этом каждый пиксель обрабатывается сам по себе, без учёта его положения относительно соседей. Это делает алгоритм простым в реализации.

Преимущества яркостного подхода:

- простота реализации;
- хорошо работает на простых изображениях с однородным фоном.

Недостатки яркостного подхода:

- сильно зависит от яркости: тени, блики и шумы мешают выделять объекты;
- не различает объекты с одинаковой яркостью, но разной формой;
- плохо справляется со сложными сценами (городские улицы, пересечённая местность), где много объектов расположены близко друг к другу. [3]

Для частичного преодоления описанных недостатков предлагается метод яркостно-метрическое разложение по топологическим признакам. В отличие от классического яркостного метода, данный алгоритм учитывает не только яркость отдельных пикселей, но и метрические характеристики - расстояния между соседними пикселями. Расстояние определяется как модуль разности их значений:

$$d = |p_1 - p_2|, \quad (1)$$

где p_1, p_2 - два соседних пикселя.

Алгоритм формирования матриц при яркостно-метрическом разложении изображения по топологическим признакам состоит из следующих шагов:

Шаг 1. Рассчитывается список расстояний между соседними пикселями. Общее количество таких расстояний для изображения размером $m * n$ равно $n(m-1) + m(n-1)$.

Шаг 2. Сортируется список расстояний по возрастанию значений.

Шаг 3. При $d \leq 0$ в компоненты объединяются пиксели с одинаковой яркостью. Изображение не меняется.

Шаг 4. При $d = 1, 2, \dots, 255$ объединяются компоненты с расстоянием $\leq d$. Поглощаемая компонента меняет цвет на цвет поглощающей, после чего расстояния пересчитываются, а список обновляется.

Шаг 5. Для каждого d ($d = 0, 1, 2, \dots, 255$) получаем изображение $I \leq d$, тем самым, для каждого d получим свой набор матриц разложения.

Ключевое отличие от яркостного метода в том, что при увеличении параметра d исходное изображение становится более однотонным. Компоненты сливаются: поглощаемая компонента меняет свой цвет на цвет поглощающей компоненты, после чего расстояния между компонентами пересчитываются заново. При слиянии приоритет получает компонента с

большим количеством точек, что позволяет отслеживать более крупные по площади объекты и игнорировать мелкие детали и шумы. [2]

Таблица 1 - Сравнение яркостного и яркостно-метрического алгоритма

Критерий	Яркостный алгоритм	Яркостно-метрический алгоритм
Учитываемые характеристики	Только яркость пикселей	Яркость + расстояния между соседними пикселями
Как строятся компоненты	По порогам яркости T (от 0 до 255)	По расстояниям d между пикселями (от 0 до 255)
Что получается в итоге	Матрицы $J_1...J_q$	Матрицы $J_1...J_q$ + дополнительные матрицы $H_1...H_f$ (устойчивость компонент)
Сложность анализа	Простой, рассматривает пиксели изолированно	Более сложный, учитывает пространственные взаимосвязи
Выделение границ	Часто неточное на сложных формах	Более точное, особенно для близко расположенных объектов
Различение объектов с одинаковой яркостью	Невозможно	Возможно по форме и расположению
Устойчивость к шумам и теням	Низкая	Выше за счёт анализа структуры

Литература

1. Еремеев С.В., Абакумов А.В., Андрианов Д.Е., Титов Д.В. Метод разложения изображения по топологическим признакам [Электронный ресурс]: URL: <https://www.elibrary.ru/item.asp?id=49814459> (дата обращения 30.11.2025)

2. Еремеев С.В., Абакумов А.В., Андрианов Д.Е., Ширабакина Т.А. Метод векторизации спутниковых снимков на основе их разложения по топологическим особенностям [Электронный ресурс]: URL: <https://www.elibrary.ru/item.asp?id=50160664> (дата обращения: 20.11.2025)

3. Еремеев С.В., Абакумов А.В. Классификация объектов на изображениях с учетом искажений на основе двухэтапного топологического анализа [Электронный ресурс]: URL: <https://cyberleninka.ru/article/n/klassifikatsiya-obektov-na-izobrazheniyah-s-uchetom-iskazheniy-na-osnove-dvuhetapnogo-topologicheskogo-analiza/viewer> (дата обращения: 27.11.2025)

Чембаров Я.Д.

Руководитель: к.т.н. доцент кафедры ИС Канунова Е.Е.

Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: kanunovae@list.ru

Разработка распределённой сети обмена файлами на основе Yggdrasil-Network

В современных условиях развития информационных технологий вопросы обеспечения приватности, устойчивости к цензуре становятся приоритетными. Традиционные клиент-серверные архитектуры уязвимы к отказам и внешним ограничениям, что стимулирует интерес к децентрализованным решениям. Отсутствие единой точки отказа главное преимущество сетей ячеистой топологии. Так Yggdrasil-Network предоставляет сквозное шифрование и IPv6-адресацию для реализации отказоустойчивой сети. В рамках работы разработана распределённая система обмена файлами, функционирующая в оверлейной сети.

Программный комплекс реализован на языках C++ и Python, что позволило совместить высокую производительность компонентов работающих с сетью с гибкостью инструментов подготовки данных. Система имеет кроссплатформенную архитектуру, поддерживающую работу в операционных системах семейств Windows NT и Unix-подобных. Взаимодействие между узлами осуществляется по протоколу IPv6.

Система состоит из нескольких функциональных модулей:

1. Средство обнаружения узлов: реализует механизм обмена адресами активных участников сети. Клиент запрашивает списки известных узлов у серверов, формируя базу соседей, позволяя сети организовываться без участия центральной точки отказа.

2. Модуль файлообмена: осуществляет непосредственную передачу данных между пирами. Система поддерживает работу с манифестами — метафайлами содержащими информацию о задаче. Поиск файла происходит через обход узлов, указанных в манифесте и его обновляемых копиях.

3. Средство оповещения о сидировании: реализует процесс обновления манифеста. После проверки целостности данных узел сообщает другим участникам сети о готовности стать новым источником задачи.

4. Инструменты подготовки и проверки: работают с архивами, контрольными суммами.

Ряд особенностей проекта - манифеста используется для расширения списка адресов сидеров. Инкапсуляция всей логики с IPv6-сокетами в классы Server и Client обеспечивает чёткое разграничение полномочий между компонентами.

Интерфейс выполнен в виде консольных утилит с использованием конфигурационных файлов. Разработанное решение позволяет построить распределённую сеть файлообмена как поверх интернета так и в автономных сетях.

Литература

1. Ашарина, И. В. "Основы программирования на языках C и C++: Курс лекций для высших учебных заведений". – Москва: Гор. линия-Телеком, 2018. — 208 с.

2. Свейгарт, Эл. "Автоматизация рутинных задач с помощью Python: практическое руководство для начинающих". – Москва: Вильямс, 2016. – 592 с.

3. The BitTorrent Protocol Specification (описание работы протокола) [Электронный ресурс] // URL: https://bittorrent.org/beps/bep_0003.html (дата обращения 04.04.26).

Yggdrasil-Network Documentation (официальная документация по архитектуре и протоколам Yggdrasil) [Электронный ресурс] // URL: <https://yggdrasil-network.github.io/documentation.html> (дата обращения 04.04.26).