

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

**61-я НАУЧНАЯ КОНФЕРЕНЦИЯ АСПИРАНТОВ, МАГИСТРАНТОВ И
СТУДЕНТОВ УЧРЕЖДЕНИЯ ОБРАЗОВАНИЯ
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»**

(Минск, 21–25 апреля 2025 года)

**МАТЕРИАЛЫ КОНФЕРЕНЦИИ
по направлению 2:**

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И УПРАВЛЕНИЕ

Минск БГУИР 2025

УДК 004.9+681.5
ББК 32.97+32.965
П99

Редакционная коллегия:

Л. Ю. Шилин, Д. П. Кукин, А. В. Марков, Д. В. Шункевич, А. А. Навроцкий,
Л. В. Николаева, В. И. Журавлев, А. Ф. Трофимович, М. В. Ковалёв

П99 61-я научная конференция аспирантов, магистрантов и студентов учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» : материалы конференции по направлению 2 : Информационные технологии и управление (Минск, 21–25 апреля 2025 года) / редкол. : Л. Ю. Шилин [и др.]. – Минск : БГУИР, 2025. – 143 с. : ил.
ISBN 978-985-543-624-0.

В сборник включены доклады, представленные на 61-й научной конференции аспирантов, магистрантов и студентов высших учебных заведений.

Материалы одобрены организационным комитетом и печатаются в авторской редакции.

Адресуется аспирантам, магистрантам, студентам высших учебных заведений, научным сотрудникам, а также специалистам предприятий в сфере IT-технологий.

УДК 004.9+681.5
ББК 32.97+32.965

ISBN 978-985-543-624-0

© УО «Белорусский государственный университет информатики и радиоэлектроники», 2025

ОРГАНИЗАЦИОННЫЙ КОМИТЕТ

Председатель

Богущ В. А. ректор, д-р физ.-мат. наук, профессор

Заместители председателя:

Давыдов М. В. первый проректор, канд. физ.-мат. наук, доцент

Стемпичский В. Р. проректор по научной работе, канд. техн. наук, доцент

Рыбак В. А. проректор по учебной работе, канд. техн. наук, доцент

Ответственный секретарь

Черноокая В. С. - заведующий сектором студенческой науки

Члены оргкомитета:

Лихачевский Д. В. декан факультета компьютерного проектирования,
канд. техн. наук, доцент,
председатель комиссии по организации и проведению конференции
«Электронные системы и технологии»

Шилин Л. Ю. декан факультета информационных технологий и управления,
д-р техн. наук, профессор,
председатель комиссии по организации и проведению конференции
«Информационные технологии и управление»

Гранько С. В. декан факультета радиотехники и электроники,
канд. техн. наук, доцент, председатель комиссии по
организации и проведению конференции «Радиотехника и электроника»

Ульянов Н. И. декан факультета компьютерных систем и сетей,
канд. техн. наук, доцент, председатель комиссии по организации и
проведению конференции «Компьютерные системы и сети»

Дробот С. В. декан факультета информационной безопасности, канд. техн. наук, доцент,
председатель комиссии по организации и проведению конференции
«Инфокоммуникации»

Лаврова О. И. декан инженерно-экономического факультета, канд. экон. наук,
доцент, председатель комиссии по организации и проведению
конференции «Проблемы экономики и информационных технологий»

Колегаев В. Г. начальник военного факультета, канд. воен. наук, доцент,
председатель комиссии по организации и проведению конференции
«Инновационные технологии в учебном процессе»

Бондарик В. М. декан факультета доуниверситетской подготовки и профессиональной
ориентации канд. техн. наук, доцент, председатель комиссии по
организации и проведению конференции «Функционирование
белорусского и русского языков в условиях информатизации общества»

Маковский М. Л. зам. декана факультета компьютерных технологий
Института информационных технологий БГУИР, председатель
комиссии по организации и проведению конференции
«Информационные системы и технологии»

Куган С. Ф. начальник управления подготовки научных кадров
высшей квалификации, д-р экон. наук, доцент

Баранова М.С. заведующая Студенческой научно-исследовательской лабораторией,
заместитель председателя Совета молодых ученых БГУИР, к.ф.-м.н.

Шаталова В.В. директор, канд. техн. наук, доцент, председатель комиссии по
организации и проведению «Научной конференции учащихся
филиала БГУИР «Минский радиотехнический колледж».

Алексеева С. А. начальник управления воспитательной работы с молодежью

Комиссия конференции «Информационные технологии и управление»

Шилин Л.Ю.	декан факультета информационных технологий и управления, д-р техн. наук, профессор - председатель комиссии по организации и проведению конференции «Информационные технологии и управление»
Ковалёв М.В.	заместитель декана факультета информационных технологий и управления, ст-й преподаватель, ответственный за подготовку, выпуск программы и электронного сборника материалов конференции «Информационные технологии и управление»
Трофимович А.Ф.	заместитель декана факультета информационных технологий и управления, член редакционной коллегии сборника материалов конференции «Информационные технологии и управление»
Голенков В. В.	д-р техн. наук, профессор
Кукин Д. П.	канд. техн. наук, доцент, зав. кафедрой ВМиП
Марков А. В.	канд. техн. наук, доцент, зав. кафедрой СУ
Шункевич Д. В.	канд. техн. наук, доцент, зав. кафедрой ИИТ
Захарьев В. А.	канд. техн. наук, доцент каф. СУ
Навроцкий А. А.	канд. физ.-мат. наук, доцент, зав. кафедрой ИТАС
Николаева Л. В.	канд. истор. наук, доцент, зав. кафедрой ГД
Журавлёв В. И.	канд. техн. наук, доцент, зав. кафедрой ТОЭ
Хаджинов М. К.	канд. техн. наук, доцент кафедры СУ

Секция «Автоматизированные системы обработки информации»

Председатель:	канд. физ.-мат. наук, доцент Навроцкий А. А.
Члены жюри:	д-р тех. наук, проф. Муха В. С. канд. тех. наук, доцент Ломако А. В. канд. тех. наук, доцент Герман О. В.
Секретарь	ст. преподаватель Боброва Т. С.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ФРЕЙМВОРКОВ ДЛЯ РАЗРАБОТКИ MAS

В статье представлен сравнительный анализ фреймворков для разработки многоагентных систем с акцентом на их возможности моделирования, включая работу с расширенными сетями Петри. Даны рекомендации по выбору фреймворков в зависимости от специфики задач.

ВВЕДЕНИЕ

Многоагентные системы (MAS) представляют собой набор автономных программных агентов, взаимодействующих для решения сложных задач в таких областях, как искусственный интеллект, IoT и управление производственными процессами. Выбор подходящего фреймворка для разработки MAS определяет эффективность, масштабируемость и гибкость системы[1].

I. КРАТКИЙ ОБЗОР ФРЕЙМВОРКОВ

Для анализа выбраны пять фреймворков: JACAMO, JADE, SPADE, AnyLogic, LangGraph.

JACAMO сочетает три технологии: Jason для программирования BDI-агентов (Belief-Desire-Intention), CArtaGO для моделирования среды и Moise для организации взаимодействий. Фреймворк востребован в робототехнике и социальных симуляциях, где требуется координация множества агентов. Но его сложная архитектура требует глубокого изучения компонент.

JADE, реализованный на Java, остается промышленным стандартом благодаря поддержке FIPA-стандартов и распределенной архитектуре. Он подходит для масштабируемых систем, таких как управление логистическими цепочками или телекоммуникационными сетями, но требует знания многопоточного программирования.

SPADE, написанный на Python, выделяется простотой интеграции с веб-сервисами через протокол XMPP. Его асинхронная модель идеальна для быстрого прототипирования решений в IoT и умных городах, где важна скорость разработки.

AnyLogic поддерживает визуальное моделирование сетей Петри и гибридные симуляции, LangGraph фокусируется на графовых моделях для динамических процессов с интеграцией ИИ.

II. КЛЮЧЕВЫЕ КРИТЕРИИ СРАВНЕНИЯ ФРЕЙМВОРКОВ

При оценке учитывались: язык программирования, соответствие стандартам (FIPA, BDI),

возможности моделирования, интеграция с искусственным интеллектом, простота освоения и активность сообщества. Например, JADE, будучи FIPA-совместимым, обеспечивает надежную совместимость с другими системами.

III. СРАВНИТЕЛЬНЫЙ АНАЛИЗ

JADE, основанный на Java, предлагает высокую производительность и кроссплатформенность, но менее удобен для исследовательских задач. JACAMO, помимо гибкости BDI-модели, требует изучения специфического синтаксиса AgentSpeak. SPADE, благодаря Python и поддержкой LLM-плагинов, позволяет быстро создавать прототипы, но уступает в поддержке сложных организационных структур.

Для моделирования процессов на сетях Петри наиболее подходит AnyLogic, в нём визуальный редактор и гибридное моделирование (агенты+дискретные события). Примеры включают оптимизацию производственных линий и цепочек поставок. LangGraph служит альтернативой задач, где графы эмулируют структуру сетей Петри, а JADE в сочетании с библиотекой PetriNetAPI подходит для промышленных систем с требованиями формальной верификации.

IV. ВЫВОДЫ

JACAMO оптимален для проектов с высокой координацией агентов, таких как робототехника. JADE доминирует в промышленных распределенных системах, а SPADE незаменим для быстрого прототипирования в IoT. Для задач, связанных с сетями Петри, ключевыми инструментами становятся AnyLogic и LangGraph, тогда как интеграция MAS с ИИ (например, через Llama Agents) расширяет возможности автономного планирования[2].

1. Multi-Agent Systems / J. Vicente, B. Vicente // Applied Sciences. – 2019. – Vol. 9, № 7. – P. 1–7.
2. Comparison of Multi-Agent Platform Usability for Industrial-Grade Applications / Z. Wrona, M. Ganzha, M. Paprzycki [et al.] // Applied Sciences. – 2024. – Vol. 14, № 22. – P. 1-28.

Агель Артём Александрович, студент ФИТиУ БГУИР, agel824@gmail.com.

Пашковец Матвей Вячеславович, студент ФИТиУ БГУИР, matvey.pashkovez01@gmail.com.

Научный руководитель: Хаджинова Наталья Владимировна, старший преподаватель кафедры ИТАС БГУИР, kha.jynova@bsuir.by.

АВТОМАТИЗИРОВАННАЯ СИСТЕМА ГЕНЕРАЦИИ ВОПРОСОВ

Разработана автоматизированная система генерации вопросов, что позволяет на основе загружаемого пользователем текста формировать набор вопросов, соответствующих содержанию материала.

ВВЕДЕНИЕ

В условиях активной цифровизации образовательного процесса особенно актуальной становится задача разработки проверочных материалов. Самостоятельное составление тестовых вопросов для оценки знаний часто требует значительных ресурсов, связанных с анализом учебного материала и разнообразием формулировок [1]. Развитие технологий обработки естественного языка (NLP) предоставляет новые возможности для автоматизации рутинных образовательных процессов, включая генерацию проверочных материалов [2]. Для решения поставленной задачи была разработана автоматизированная система на основе языковой модели DeepSeek, способная генерировать вопросы для контроля знаний на базе произвольного текста.

I. ВЫБОР ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ДЛЯ РЕАЛИЗАЦИИ ПРОГРАММНОГО РЕШЕНИЯ

Для разработки программного обеспечения был выбран язык Python. Для создания графического интерфейса была использована библиотека Tkinter. Для обработки различных типов файлов применялись библиотеки python-docx, pdfplumber, python-pptx, Pillow и pytesseract. Генерация учебных вопросов осуществляется с использованием большой языковой модели DeepSeek. Взаимодействие с моделью реализовано через DeepSeek API. Для интеграции с API используется совместимая с ним библиотека openai [3].

Архитектура системы разработана с акцентом на модульность, что обеспечивает гибкость в разработке и возможность масштабирования. Система логически разделена на клиентскую и серверную части, соединённые стандартизированным интерфейсом передачи данных.

Такая структура упрощает расширение функциональности, включая добавление поддержки новых форматов, типов вопросов, языков, а также интеграцию с внешними образовательными платформами.

Арсенович Павел Александрович, студент кафедры информационных технологий автоматизированных систем БГУИР, arsenpashmail@gmail.com.

Земляник Руслана Владимировна, студентка кафедры информационных технологий автоматизированных систем БГУИР, rzemlanik@gmail.com.

Научный руководитель: Шилин Леонид Юрьевич, декан факультета информационных технологий и управления БГУИР, доктор технических наук, профессор, dekfutu@bsuir.by.

II. ОПИСАНИЕ АЛГОРИТМА РАБОТЫ СИСТЕМЫ

Алгоритм работы системы построен следующим образом:

1. Загружается файл с учебным материалом.
2. Выполняется проверка формата и содержимого файла.
3. Извлекается и отображается текстовое содержимое в редакторе.
4. Текст вместе с промптом передаётся в языковую модель через API.
5. Производится семантический анализ и генерация вопросов с учётом заданных параметров.
6. Результаты отображаются для редактирования, копирования и экспорта.

В случае возникновения ошибки система информирует пользователя о возникшей проблеме, предоставляет рекомендации по их исправлению.

III. ЗАКЛЮЧЕНИЕ

Была разработана система для генерации вопросов на основе методических материалов, предназначенная для оценки знаний студентов по учебной дисциплине "Автоматизированное проектирование электрических цепей". В рамках тестирования был сгенерирован пакет вопросов, проведено тестирование на группе студентов-добровольцев. Полученные результаты подтверждают эффективность разработанной системы.

1. G. Kurdi, J. Leo, B. Parsia, U. Sattler, and S. Al-Emari International Journal of Artificial Intelligence in Education, 30(1):121–204, 2020.
2. Zhang D., Zhao W., Wang Y., et al. Automatic question generation from text: A survey of algorithms and applications // ACM Computing Surveys. – 2022. – Vol. 54. – Iss. 4. – P. 78.
3. DeepSeek API Documentation [Электронный ресурс]. – Режим доступа: <https://api-docs.deepseek.com>. – Дата доступа: 08.04.2025.

ПРИНЯТИЕ РЕШЕНИЙ И ТЕОРИЯ ИГР. ОСНОВНЫЕ ПОНЯТИЯ, ПРОБЛЕМЫ РЕАЛИЗАЦИИ

Рассматриваются основные понятия принятия решений, их связь с теорией и методами теории игр. Приведены некоторые проблемные аспекты при реализации задач вышеуказанных теорий.

ВВЕДЕНИЕ

Теория принятия решений (ТПР) изучает закономерности выбора людьми путей решения проблем и задач, а также способы достижения желаемого результата а теория игр (ТИ) - формальные модели принятия оптимальных решений в условиях конфликта. К основным понятиям ТПР следует отнести: решение, процесс принятия решения, лицо, принимающее решение, цель принятия решения, средства реализации решения, обычно применяются методы ТИ.

I. ОСНОВНЫЕ ПОНЯТИЯ ТИ

Классическим примером в ТИ является теорема применения решения в *смешанных стратегиях* Дж. Фон Неймана

Каждую задачу можно представить в общем виде как:

$$S_a = (p_1, p_2, \dots, p_n)$$

$$S_b = (q_1, q_2, \dots, q_n)$$

p – вероятность выбора стратегий игрока А,

q – вероятность выбора стратегий игрока В.

$$\sum_{i=1}^n p_i = 1 \quad \sum_{i=1}^n q_i = 1$$

Пример: игра задана платёжной матрицей

$$P = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

Выигрыш игрока А, использует смешанную стратегию, составит $S_a = \begin{pmatrix} 1 & A_2 \\ p_1 & p_2 \end{pmatrix}$.

Выигрыш игрока В, чистая стратегия В1 (это соответствует первому столбцу платежной матрицы Р), равен цене игры v : $a_{11}p_1^* = a_{21}p_2^* = v$.

Такой выигрыш получит игрок А, если второй игрок применяет стратегию В2, т.е. $a_{12}p_1^* = a_{22}p_2^* = v$. Зная, что $p_1^* + p_2^* = 1$, составим систему уравнений для определения оптимальной стратегии S_a^* и цены игры v .

$$a_{11}p_1 + a_{21}p_2 = v$$

$$a_{12}p_1 + a_{22}p_2 = v$$

$$p_1 + p_2 = 1$$

Решив эту систему, находим оптимальную стратегию

$$p_1 = \frac{a_{22} - a_{21}}{a_{11} + a_{22} - a_{12} - a_{21}} \quad (1)$$

$$p_2 = \frac{a_{11} - a_{12}}{a_{11} + a_{22} - a_{12} - a_{21}} \quad (2)$$

$$\text{и цену игры} \quad v = \frac{a_{22}a_{11} - a_{12}a_{21}}{a_{11} + a_{22} - a_{12} - a_{21}} \quad (3)$$

Беспалов Сергей Алексеевич, магистрант кафедры информационных технологий автоматизированных систем БГУИР, bespalov@bsuir.by.

Научный руководитель: Навроцкий Анатолий Александрович, заведующий кафедрой информационных технологий автоматизированных систем БГУИР, к. физ.-мат. наук, доцент, navrotsky@bsuir.by.

Для определения оптимальной стратегии и игрока В, учитывая, что при любой чистой стратегии игрока А (A_1 или A_2) средний проигрыш игрока В равен цене игры v , т.е. система уравнений для определения стратегии S_b^* и цены игры v имеет вид:

$$a_{11}q_1 + a_{21}q_2 = v$$

$$a_{12}q_1 + a_{22}q_2 = v$$

$$q_1 + q_2 = 1$$

и оптимальная стратегия S_b^* (q_1 и q_2) определяется по формулам аналогичными (1), (2), (3) заменив параметры p_1 и p_2 на q_1 и q_2 соответственно.

II. ПРОБЛЕМЫ ПРИМЕНЕНИЯ МЕТОДОВ ТИ

ТИ затруднительно применять при множестве ситуаций равновесия. Игроки могут недостаточно или неясно информированы о стратегиях друг друга и не всегда принимают решения рационально, т.е. игрок приходит к проигрышу. Экспериментально доказано, что при расширении игры до десяти и более этапов игроки не в состоянии пользоваться соответствующими алгоритмами и продолжать игру с равновесными стратегиями.

III. ВЫВОДЫ

При анализе методов реализации ТИ и ТПР прослеживается тесная взаимосвязь между обеими теориями, примеры: дерево решений в ТПР является развернутой формой игры в ТИ, условие постановки задачи в ТПР аналогична представлению нормальной формы игры в ТИ.

ТИ и ТПР тесно связаны с теорией вероятностей и исходы игры или результат принятия решений зависят от применяемых методов реализации расчетов моделирования. Отсюда следует, что задачи ТПР успешно применяются с использованием методов ТИ.

1. Дж. фон Нейман, О. Моргенштерн. «Теория игр и экономическое поведение», Наука, 1970.
2. Оуэн Г. «Теория Игр». – М.: Мир, 1970.
3. Ларичев О.И. «Теория и методы принятия решений, а также Хроника событий в Волшебных странах». – М.: Логос, 2003.

АЛГОРИТМ ИЗМЕНЕНИЯ ЭМОЦИОНАЛЬНОЙ ОКРАСКИ АУДИОФАЙЛА

Реализован алгоритм изменения эмоциональной окраски речи. Алгоритм базируется на модели Wav2Vec2, дообученной с использованием датасета, в котором один человек озвучивает фразы с различными эмоциями. На основе полученной модели разработано графическое приложение, позволяющее преобразовывать эмоции аудиозаписей.

ВВЕДЕНИЕ

Работа посвящена алгоритму, разработанному для анализа и изменения эмоциональной окраски аудиофайлов, с применением предобученной модели Wav2Vec2.

Дообучение модели выполнялось на датасете, озвученном одним спикером с нейтральным, радостным, грустным, удивлённым и раздражённым эмоциональными окрасами речи.

I. АЛГОРИТМ РАБОТЫ

Входной аудиосигнал сначала разбивается на фиксированные аудиофреймы $x \in \mathbb{R}^M$, где M — размерность входного вектора-фрейма. Для извлечения признаков используется дообученная модель Wav2Vec2, которая преобразует каждый фрейм в скрытый вектор $h \in \mathbb{R}^N$ по формуле

$$h = \tanh(Wx + b), \quad (1)$$

где $N \in [768, 1024]$ выбирается исходя из требований к качеству и скорости. После обработки всех фреймов получается последовательность $\{h_i\}_{i=1}^T$, (T — количество фреймов в сигнале), передаваемая в блок коррекции эмоций.

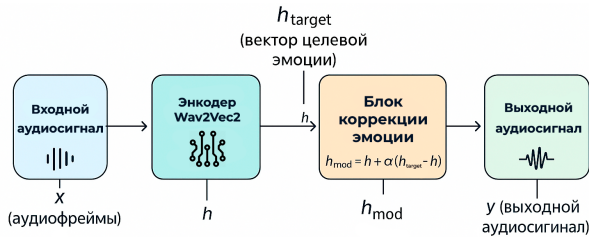


Рис. 1 – Схема работы системы

Исходное представление h смещается в направлении вектора целевой эмоции h_{target} , вычисляемого как

$$h_{\text{target}} = \frac{1}{K} \sum_{j=1}^K h_j^{(\text{target})}. \quad (2)$$

Василевский Владислав Валерьевич, студент кафедры информационных технологий автоматизированных систем, БГУИР, vlad.vasilevskiy.07@gmail.com.

Рында Роман Дмитриевич, студент кафедры информационных технологий автоматизированных систем, БГУИР, romarynda1@gmail.com.

Научный руководитель: Езовит Алексей Владимирович, ассистент кафедры информационных технологий автоматизированных систем, БГУИР, a.ezovit@bsuir.by.

Далее формируется дельта-вектор $d = h_{\text{target}} - h$, который проходит через нейронную сеть построенную на архитектуре MLP (multilayer perceptron):

$$d' = W_2 \text{ReLU}(W_1 d + b_1) + b_2, \quad (3)$$

где $W_1 \in \mathbb{R}^{K \times N}$ и $W_2 \in \mathbb{R}^{N \times K}$. Полученный вектор масштабируется пользовательским коэффициентом $\alpha \in [0, 1]$ и складывается с исходным представлением: $h_{\text{mod}} = h + \Delta$, $\Delta = \alpha d'$.

Для обеспечения плавности во времени к $\{h_{\text{mod},i}\}$ дополнительно применяется нормализация и, при необходимости, одномерная свёртка.



Рис. 2 – Внутренняя логика блока коррекции эмоций

Последовательность модифицированных векторов $\{h_{\text{mod},i}\}$ подаётся на вход вокодера, который синтезирует выходной аудиосигнал y . На финальном этапе выполняется нормализация амплитуды и сглаживание возможных артефактов, полученных в процессе преобразования.

II. ВЫВОДЫ

Предложен алгоритм, построенный на основе дообученной модели Wav2Vec2 и модуле коррекции эмоций, позволяющий изменять эмоциональную окраску аудиофайлов. Разработанная модель интегрирована в приложение, построенное на базе фреймворка qt.

1. Baevski, A., Zhou, H., Mohamed, A., Auli, M. Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations // Advances in Neural Information Processing Systems. – 2020.
2. Eyben, F., Wöllmer, M., Schuller, B. Introduction to the special issue on affective computing // IEEE Transactions on Affective Computing. – 2016.

АВТОМАТИЧЕСКОЕ РАСПОЗНАВАНИЕ АВТОМОБИЛЬНЫХ НОМЕРОВ

Рассматривается задача автоматического распознавания автомобильных номеров. Сравнивается различное ПО для их выделения и распознавания. Оценивается влияние аугментации данных и предобработки изображений.

ВВЕДЕНИЕ

Автоматическое распознавание номерных знаков транспортных средств (Automatic License Plate Recognition, ALPR) представляет собой важное направление компьютерного зрения с широким спектром практических применений. Процесс ALPR включает несколько этапов обработки изображения: точная локализация номерной пластины, сегментация символов, распознавание текстовой информации [1].

I. ВЫДЕЛЕНИЕ ОБЛАСТИ НОМЕРА ПРИ ПОМОЩИ YOLO

Для эффективного решения задачи необходимо учитывать, поддерживает ли используемое ПО автоматическую локализацию номерной пластины. Для данной подзадачи была обучена YOLOv8 — нейросетевая модель object detection. Модель была обучена на датасете, включающем 141 изображение автомобилей с белорусскими номерами. Во втором эксперименте модель была обучена на аугментированном датасете из 987 синтетических изображений с имитацией погодных условий: дождь, снег, изменение освещенности. Обе версии модели продемонстрировали высокую эффективность в локализации номерных пластин — до 98 и 100% соответственно.

II. СРАВНЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РАСПОЗНАВАНИЯ ТЕКСТА НОМЕРОВ

Для задачи оптического распознавания текста (OCR) были выбраны три open-source решения для Windows 11: EasyOCR 1.6.2 (нейросетевая библиотека на PyTorch), Tesseract 5.5.0 (OCR-ПО от Google) и OpenALPR 2.3.0 (ALPR-система, включающая OCR и локализацию пластин). На данном этапе все системы использовались без дообучения на пользовательских данных. Попытка дообучить Tesseract завершилась неудачно из-за некорректной генерации .box-файлов — сегментация символов автоматически выполнялась с искажениями. Примеры распознавания демонстрируются на рис.1. Было проведено сравнение эффективности OCR-систем на исходных и предобработанных изображениях.

Грабовский Алексей Кириллович, студент кафедры информационных технологий автоматизированных систем БГУИР, aleshamayg@gmail.com.

Научный руководитель: Навроцкий Анатолий Александрович, заведующий кафедрой информационных технологий автоматизированных систем БГУИР, доцент, navrotsky@bsuir.by.

YOLO+EasyOCR	YOLO+Tesseract	OpenALPR
BY 0402 BC-1 ✓ – оригинальный 0402 BC-1 ✓ – обработанный	0402BC1 ✓ – оригинальный Не распознано ✗ – обработанный	0402BC1 ✗ – оригинальный Не распознано ✗ – обработанный
BY 0518HB-5 ✗ – оригинальный BY 0518H8-5 ✗ – обработанный	0518HB5 ✓ – оригинальный F0518HB5 ✗ – обработанный	0518H ✗ – оригинальный 0518HB5 ✓ – обработанный
Не распознано ✗ – оригинальный 81 000 ✗ – обработанный	90017 ✗ – оригинальный Не распознано ✗ – обработанный	Не распознано ✗ – оригинальный 00010P5 ✗ – обработанный
6165HA-4 ✓ – оригинальный 6165HA-4 ✓ – обработанный	6165HA4 ✓ – оригинальный 6165HA4 ✓ – обработанный	Не распознано ✗ – оригинальный 65HA4 ✗ – обработанный

Рис. 1 – Примеры результата работы ПО

В подвыборках с наиболее чёткими номерами доля **полностью** правильно распознанных номеров у OpenALPR (open-source версии) достигала 40%, в то время как у EasyOCR и Tesseract — около 30%. При этом предобработка (контрастная бинаризация) не оказала однозначно положительного или отрицательного влияния на качество распознавания. Эффект варьировался в зависимости от конкретной модели и изображения.

III. ВЫВОДЫ

Полученные результаты демонстрируют, что ALPR-задачи могут быть решены исключительно с использованием open-source инструментов. Аугментация изображений не оказывает большого эффекта на итоговые результаты распознавания. Эффект от предобработки изображения остаётся ограниченным и требует дальнейшей оптимизации, равно как и устойчивость подобных подходов в целом, особенно при наличии каких-либо искажений изображения.

1. ResearchGate [Электронный ресурс]. Режим доступа: https://www.researchgate.net/publication/387137009_Primenenie_komputernogo_zreniya_dla_raspoznaniya_avtomobilnyh_nomerov – Дата доступа: 13.04.2025.

ОПТИМИЗАЦИЯ РАБОТЫ КОПИРОВАЛЬНОГО ЦЕНТРА

Изучены современные методы работы мессенджером Telegram. Рассмотрены методы оптимизации взаимодействия пользователя и работников копировального центра. Описаны этапы работы с ботом и способы оформления заказа.

Современные условия цифровизации требуют от сервисных предприятий внедрения автоматизированных решений для повышения эффективности работы. В докладе представлен алгоритм работы телеграм-бота, предназначенного для управления заказами в копировальном центре. Использование телеграм-бота позволяет исключить «ручную» обработку заказов, задержки и ошибки, обусловленные человеческим фактором [1].

Разработка бота выполнена на языке Python с использованием библиотеки aiogram для работы с API Telegram и sqlalchemy для управления базой данных SQLite. Асинхронное программирование, реализованное через aiogram, обеспечивает высокую производительность при обработке множества запросов. Архитектура системы минимизирует нагрузку на сервер, что делает решение устойчивым к переменному потоку пользователей. Выбор технологий обусловлен их надежностью и поддержкой масштабируемости [2].

Бот предоставляет удобный интерфейс для управления заказами. После ввода команды «start» пользователю доступна инструкция по работе с системой. Основной функционал включает: просмотр баланса с автоматическим списанием стоимости услуг по принципу предоплаты; реферальную систему, стимулирующую привлечение новых клиентов (пригласивший получает процент от заказов, а новый пользователь — скидку); оформление и отмену заказов на печать, что исключает нежелательные расходы и снижает затраты на персонал.

Инновационность решения заключается в интеграции реферальной системы и использовании

асинхронного подхода для повышения производительности.

Тестирование подтвердило устойчивость системы к нагрузкам. При минимальном числе запросов потребление ресурсов составило 40%, а в условиях пиковой нагрузки бот сохранял стабильность без сбоев. Эти данные свидетельствуют о пригодности решения для реальной эксплуатации в копировальном центре.

Внедрение бота позволяет отказаться от использования колл-центра, снизить затраты, ускорить обработку заказов и уменьшить ошибки. Рост удовлетворенности клиентов подтверждается увеличением повторных обращений и притоком новых пользователей через реферальную систему.

Разработанное решение является эффективным инструментом для оптимизации бизнес-процессов. Успешное тестирование и внедрение подчеркивают практическую значимость работы.

Созданный телеграм-бот решает поставленные задачи и открывает возможности для развития. Интеграция с облачными сервисами обеспечит масштабируемость, что позволит применять решение в крупных копировальных центрах [3].

1. Оптимизация бизнес-процессов [Электронный ресурс]. – Режим доступа: <https://hsbi.hse.ru/articles/optimizatsiya-biznes-protsessov/>. – Дата доступа: 29.03.2025.
2. Документация aiogram [Электронный ресурс]. – Режим доступа: <https://docs.aiogram.dev/en/v2.25.1/>. – Дата доступа: 29.03.2025.
3. Документация для фреймворка fastapi [Электронный ресурс]. – Режим доступа: <https://fastapi.tiangolo.com/>. – Дата доступа: 29.03.2025.

Иосько Михаил Алексеевич, студент кафедры информационных технологий автоматизированных систем БГУИР, ioskomihailaa@gmail.com.

Научный руководитель: Навроцкий Анатолий Александрович, заведующий кафедрой информационных технологий автоматизированных систем БГУИР доцент, navrotsky@bsuir.by.

ЭТИЧЕСКИЕ АСПЕКТЫ ПРИНЯТИЯ РЕШЕНИЙ В АВТОНОМНЫХ MAS

Автономные многоагентные системы (MAS) становятся ключевым элементом современных технологий — от умных городов до медицинской диагностики. Однако их внедрение сталкивается с этическими дилеммами, возникающими при противоречии между эффективностью (максимизацией общей полезности) и справедливостью (равномерным или приоритетным распределением ресурсов). В исследовании предложены гибридные модели, интегрирующие утилитаристские и эгалитарные подходы с использованием адаптивных весов и динамических приоритетов.

ВВЕДЕНИЕ

Развитие искусственного интеллекта и MAS приводит к активному применению автономных агентов в критически важных сферах: логистике, здравоохранении, финансах, управлении городским трафиком. Алгоритмы таких систем зачастую оптимизируют технические параметры (скорость, точность), игнорируя многокритериальность этических решений. Вопрос ответственности за несправедливость (разработчик, оператор или сама система) остаётся открытым и требует внедрения механизмов этического аудита и объяснимости (XAI).

МЕТОДОЛОГИЯ

Исследование базируется на симуляциях в условиях ограниченных ресурсов с использованием Python и платформы Mesa. Рассмотрены кейс-стади: управление трафиком беспилотников, распределение бюджетов в социальных программах и медицинское распределение донорских органов. Количественная оценка проводится с помощью таких метрик, как индекс Джини, индекс Джайнса и время отклика.

I. ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ЭТИЧЕСКИХ ДИЛЕММ

Обзор литературы выявил, что утилитарный подход, направленный на максимизацию общей пользы, может приводить к маргинализации меньшинств (например, алгоритмы кредитного скоринга). Роулсианская «справедливость как честность» требует установления минимальных стандартов для всех, что зачастую снижает эффективность системы. Таким образом, необходимы адаптивные модели, способные менять приоритеты в зависимости от ситуации.

II. КРИТИКА СУЩЕСТВУЮЩИХ МОДЕЛЕЙ

Фиксированные квоты и жесткие правила нарушают гибкость MAS, а обучение на исторических

данных часто воспроизводит прошлые предубеждения. Эксперимент «The Trolley Problem» или «дилеммы трамвая» в MAS, проведенный на базе децентрализованных решений в управлении беспилотниками, показал, что децентрализация увеличивает справедливость, но замедляет реакцию системы.

III. ГИБРИДНЫЕ МОДЕЛИ БАЛАНСА

Предложены модели с динамическим перераспределением значимости критериев (адаптивные веса) и многоуровневая архитектура: на локальном уровне происходит оптимизация эффективности, а на глобальном — аудит на соответствие этическим нормам [1]. Экспериментальные данные демонстрируют, что в сценарии распределения донорских органов гибридная модель значительно снизила дискриминацию пациентов с редкими группами крови при минимальной потере эффективности. В управлении городским трафиком внедрение динамических приоритетов снизило задержки на окраинах почти наполовину без ухудшения ситуации в центральных зонах.

IV. ВЫВОДЫ

Разработка этических алгоритмов для автономных MAS требует интеграции формальных моделей распределения с адаптивными механизмами, обеспечивающими баланс между эффективностью и справедливостью. Применение гибридных моделей, основанных на количественных и качественных метриках, позволяет минимизировать социальные риски, повысить прозрачность и устойчивость систем.

1. Jiang J., Lu Z. Learning fairness in multi-agent systems // Proceedings of the 33rd International Conference on Neural Information Processing Systems. NeurIPS 2019. December 2019. P. 13854–13865. Article No.: 1241.

Кадлубай Вера Александровна, студентка ФИТиУ БГУИР, tylyla2004@gmail.com.

Научный руководитель: Хаджинова Наталья Владимировна, старший преподаватель кафедры ИТАС БГУИР, khajynova@bsuir.by.

DATA CLEANING: CHALLENGES AND MODERN APPROACHES

Data cleaning is a critical stage in data preprocessing, directly impacting the quality of analytics and machine learning models. This article examines key challenges in data cleaning—missing values, outliers, duplicates, and inconsistencies—and evaluates modern automated approaches, including rule-based systems, ML-based anomaly detection, and hybrid frameworks.

INTRODUCTION

Modern datasets suffer from incompleteness, noise, and heterogeneity due to diverse sources including IoT devices, user inputs, and legacy systems. According to Gartner (2023), poor data quality costs enterprises 15–25% of their annual revenue. This work systematically examines current challenges and analyzes cutting-edge tools for scalable data cleaning.

METHODOLOGY

The study evaluates three data cleaning approaches: rule-based (SQL/regex), ML-based (isolation forests/NLP), and hybrid (human-in-the-loop with active learning). Performance was assessed using precision/recall, runtime, and scalability on UCI/Kaggle datasets (10K–1M records).

I. KEY CHALLENGES

The field of data cleaning faces several significant challenges that require different solution approaches. Missing data presents a complex problem where the mechanism behind the missingness determines the appropriate imputation strategy, ranging from simple mean substitution to advanced predictive modeling techniques. Outlier detection must account for context-dependent anomalies, balancing statistical approaches like the 3σ rule against domain-specific thresholds. Inconsistencies in data formats, such as variations in date representations or entity naming, require sophisticated resolution methods.

II. MODERN APPROACHES

Contemporary data cleaning solutions fall into three main categories. Automated tools offer cluster-based deduplication using Levenshtein distance metrics, while TensorFlow Data Validation provides robust statistical drift detection capabilities [1]. Machine learning methods have introduced innovative solutions such as generative models (GANs) for synthetic data imputation and transformer-based frameworks like DataBERT for semantic inconsistency resolution. The most promising results come from hybrid systems that combine deterministic rules with

probabilistic machine learning, achieving an impressive 92% F1-score in industrial applications according to recent case studies.

III. CASE STUDY: TELECOM DATA CLEANING

A telecom dataset (500,000 records from 2019–2023 CRM systems) with 12% missing values, 8% duplicates, and 15% format inconsistencies was processed through an automated pipeline. Initial analysis used *Pandas Profiling* for missing data patterns and histogram-based outlier detection (e.g., call durations >24h). *Great Expectations* enforced standardization rules for phone numbers (E.164 format) and email validation (regex + DNS checks). Deduplication employed fuzzy matching (Levenshtein distance ≤ 2) with active learning on 0.5% ambiguous cases.

The automated approach reduced processing time by 58% (7.6h $\rightarrow$ 3.2h) versus manual cleaning, while improving duplicate detection by 34% and reducing false positives from 9% to 4.7%. The cleaned data enhanced churn prediction accuracy by 22% AUC. Key findings showed rule-based methods excel at format standardization, while ML requires careful labeling for edge cases. Docker containers proved valuable for pipeline reproducibility.

IV. CONCLUSIONS

Automated data cleaning solutions have demonstrated their ability to significantly reduce operational costs while improving data quality and analytical reproducibility. Future research directions include the development of self-adaptive pipelines incorporating reinforcement learning techniques. Significant challenges remain, particularly in the areas of privacy-preserving cleaning methods that comply with regulations like GDPR and the development of solutions capable of cross-domain generalization.

1. Abedjan, Z. (2016). *Data Cleaning with ML*. ACM Computing Surveys.

Vera Kadlubai, student of FITC BSUIR, tylyla2004@gmail.com.

Alexey Trofimovich, Senior Lecturer of ITAS Department, FITC, BSUIR, trofimaf@bsuir.by

ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ В ЗАКАЗНОЙ РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Анализируется роль бизнес-аналитика в аутсорсинге: преимущества (независимая оценка, опыт из других проектов) и проблемы (отсутствие доступа к пользователям, неполные данные о стейкхолдерах). Подчеркивается важность адаптивности для минимизации рисков и успешной разработки.

ВВЕДЕНИЕ

Бизнес-аналитик взаимодействует с заказчиками, командой разработчиков и представителями различных отделов, проводит анализ конкурирующих приложений, изучает рынок, проводит опросы, строит модели пользовательских сценариев и прочее.

I. ОСОБЕННОСТИ АУТСОРСИНГОВОЙ РАЗРАБОТКИ

Аутсорсинг представляет собой делегирование второстепенных функций компании специализированному стороннему подрядчику. Аутсорсинг бизнес-аналитиков позволяет компаниям повысить качество и эффективность своей основной деятельности, а также освободить человеческие и финансовые ресурсы бизнеса.

Жизненный цикл разработки продукта в аутсорсинге исключает обратную связь, а фидбек от заказчика спустя значительное время после внедрения решения редкость. В связи с этим риск незамеченных во время разработки ошибок возрастает, а вероятность их повторения в будущем не уменьшается. Поэтому бизнес-аналитику необходимо постоянно пополнять опыт, анализируя свои и чужие кейсы, ведь это напрямую влияет на качество предоставляемой работы и успех продукта на рынке.

В аутсорсинге у аналитика, как и в продукте, есть возможность влиять на продукт, особенно это касается случаев, когда аутсорсинг команда глубоко погружена в продукт заказчика и долго над ним работает. Однако это влияние будет заметно менее ощутимо, чем в продукте, так как заказчик приходит с готовой идеей, а бизнес-аналитику нужно этот запрос детализировать и перевести в задачи, понятные для команды разработки. Бизнес-аналитик детализирует готовые гипотезы, согласовывает с клиентом и передает на команду разработки.

Главное преимущество бизнес-аналитика на аутсорсинге – объективная и независимая оценка ситуации в компании, анализ слабых мест и «свежий» взгляд со стороны. Необходимо учитывать опыт работы в других продуктах, что также положительно влияет на результат работы бизнес-

аналитика в аутсорсинге, так как он мог работать в подобной или той же сфере, а значит, способен принимать более квалифицированное и грамотное решения для бизнеса и компании.

Ключевым аспектом становится адаптивность стратегий: агенты корректируют свои действия на основе наблюдаемого поведения других участников и частичной информации о состоянии среды.

Для формализации конфликтов применяется аппарат Байесовых игр, где неполнота данных компенсируется вероятностными оценками намерений оппонентов. Это создаёт основу для прогнозирования конфликтных ситуаций и выбора оптимальных траекторий движения.

II. ПРОБЛЕМЫ ПРИ РАЗРАБОТКЕ В АУТСОРСИНГЕ

Популярной проблемой бизнес-аналитика в аутсорсинге является отсутствие доступа к конечным пользователям (50% случаев) и неизвестность полного перечня стейкхолдеров (57,1%). Это напрямую влияет на результат разработки продукта, так как при недовольстве одного или нескольких стейкхолдеров, внезапно появившихся в процессе разработки, команда может понести большие потери ресурсов. Также проблемой является быстрая смена приоритетов у заказчиков (35%), что обусловлено желанием успеть за релизом новых функций конкурентов.

III. ВЫВОДЫ

Бизнес-аналитик в аутсорсинге углубляет свою экспертность, способствуя росту как специалиста в различных областях разработки продуктов, работает с требованиями заказчика, что является одной из ключевых задач, которая должна выполняться на высоком уровне. Процессы должны быть детально изучены, и в аутсорсинге это проще, потому что процессы реже меняются.

1. Карл Вигерс, Джой Битти «Разработка требований к программному обеспечению» 3-е изд., Русская Редакция, 2014, 718с

Кононцева Диана Владимировна, студентка ФИТиУ БГУИР, diana.konontseva@mail.ru.

Научный руководитель: Мельник Анастасия Александровна, старший преподаватель кафедры ИТАС БГУИР, a.melnik@bsuir.by.

ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ПРОДУКТОВОЙ РАЗРАБОТКЕ

Исследуется роль бизнес-аналитика в продуктовой разработке: сбор требований, анализ рынка, взаимодействие со стейкхолдерами. Ключевые проблемы: отсутствие стратегии, частая смена приоритетов, сложность фидбэка. Подчеркивается важность адаптивности для успеха продукта.

ВВЕДЕНИЕ

Продуктовая разработка – процесс охватывающий весь жизненный цикл продукта (системы, приложения). Бизнес-аналитик в продуктовой разработке занимает одну из ключевых ролей в развитии продукта, так как именно он занимается постоянным исследованием уязвимостей и слабых мест, собирает метрики и требования, исследует рынок и конкурентов, опрашивает стейкхолдеров и работает над улучшением качества продукта, обеспечивая устойчивое развитие.

I. ОСОБЕННОСТИ ПРОДУКТОВОЙ РАЗРАБОТКИ

Ключевым в работе бизнес-аналитика в продуктовой разработке является то, что он присутствует не только при основной разработке продукта, но и при дальнейших доработках. Аналитик взаимодействует со командами продаж, поддержки, разработки, их связь напрямую влияет на успех проекта.

Глубокий анализ помогает понять жизнеспособность новых идей и осознать все потенциальные подводные камни.

Процессы, затрагиваемые бизнес-аналитиком в продуктовой разработке, имеют важное влияние на развитие проекта:

1. Сбор справочной информации: прежде всего, команда бизнес-аналитиков собирает всю информацию о текущей ситуации в сфере: последние тенденции, новейшие технологии и т.д. Особое внимание они уделяют внешним обстоятельствам, которые могут повлиять на проект.

2. Определение заинтересованных сторон: бизнес-аналитики выявляют всех потенциальных стейкхолдеров, т.е. тех, кто заинтересован или может быть заинтересован в вашем проекте и может непосредственно повлиять на него.

3. Проведение презентации и сбора фидбэка со стороны заинтересованных сторон, в том числе команды разработки.

4. Внесите изменения в документацию.

5. Сопровождение изменений, пока им не начнут следовать все.

II. ОСНОВНЫЕ ПРОБЛЕМЫ ПРОДУКТОВОЙ РАЗРАБОТКИ

Важной проблемой продуктовой разработки является отсутствие стратегии развития в явном виде (в 52% случаев), что усложняет процессы формирования и принятия решений. Еще одной проблемой является быстрая смена приоритетов и желаний стейкхолдеров (52%), что зачастую связано с погоней за конкурентами, из-за чего появляется проблема слепого копирования функционала. Это может привести к провалу продукта на рынке по причине отсутствия преимуществ относительно других, более популярных решений. Очередной проблемой можно вынести сложность налаживания фидбэка от пользователей (40% случаев). Это напрямую влияет на построение дальнейших стратегий развития продукта компании. Отсутствие отзывов пользователей влечет за собой возможные потери компании и долгий анализ причин ухудшения ситуации.

III. ВЫВОДЫ

Важная часть работы бизнес-аналитика в продукте – способность быстро реагировать на изменения. Он может долго согласовывать требования, отдать новые функции в разработку, но ситуация на рынке резко изменится. В таком случае в продукте важно быстро реагировать и корректировать план разработки.

1. Карл Вигерс, Джой Битти «Разработка требований к программному обеспечению» 3-е изд., Русская Редакция, 2014, 718с

Кононцева Диана Владимировна, студентка ФИТиУ БГУИР, diana.konontseva@mail.ru.

Научный руководитель: Мельник Анастасия Александровна, старший преподаватель кафедры ИТАС БГУИР, a.melnik@bsuir.by.

АЛГОРИТМ МАСШТАБИРОВАНИЯ ГРАФИЧЕСКИХ ЭЛЕМЕНТОВ: ПРИНЦИПЫ И ПРИМЕРЫ

В данной статье будут рассмотрены основные принципы алгоритмов масштабирования и приведен пример модификации кода, реализующего такой алгоритм.

ВВЕДЕНИЕ

Масштабирование графических элементов – это процесс изменения размеров объектов в рамках графического интерфейса или изображения. Этот процесс часто используется в разработке программного обеспечения, 3D-моделировании и в играх для адаптации элементов под разные размеры экранов и разрешения.

I. ПРИНЦИПЫ МАСШТАБИРОВАНИЯ

Основные принципы масштабирования включают:

1. **Аффинные преобразования:** Масштабирование может быть представлено аффинными преобразованиями, которые включают изменения размеров, вращения и перенос объектов.
2. **Пропорциональность:** Пропорциональное масштабирование сохраняет соотношения сторон объекта. Это важно для предотвращения искажений.
3. **Центр масштабирования:** Масштабирование может происходить относительно различных точек, таких как центр объекта, угол или произвольная точка.
4. **Растривание и векторная графика:** Для растровой графики масштабирование может повлечь за собой потерю качества, в то время как векторные изображения могут масштабироваться без потери четкости.

II. ПРИМЕР КОДА

Рассмотрим пример на Python с использованием библиотеки Pygame, которая позволяет работать с графическими элементами. В этом примере создается простой графический элемент (прямоугольник) и реализуется его масштабирование. Сначала инициализируется библиотека Pygame и создается окно заданного размера. Устанавливаются начальные размеры и цвет прямоугольника.

```
pygame.init()
screen_width = 800
screen_height = 600
screen = pygame.display.set_mode(
    (screen_width, screen_height))
rectangle_color = (0, 128, 255)
rectangle_width = 100
rectangle_height = 50
scale_factor = 1.0
```

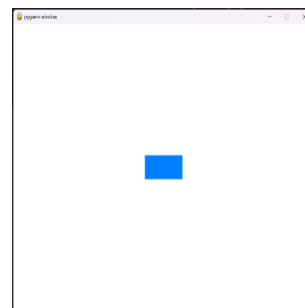
Главный цикл программы. В этом цикле обрабатываются события, такие как нажатие клавиш.

```
while True:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()
            sys.exit()
        if event.type == pygame.KEYDOWN:
            if event.key == pygame.K_UP:
                scale_factor += 0.1
            elif event.key == pygame.K_DOWN:
                scale_factor = max(0.1, scale_factor - 0.1)
```

Далее происходит масштабирование. При нажатии клавиш «вверх» и «вниз» изменяется коэффициент масштабирования и рассчитываются новые размеры прямоугольника. Очищается экран и рисуется новый масштабированный прямоугольник.

```
new_width = int(rectangle_width * scale_factor)
new_height = int(rectangle_height * scale_factor)
screen.fill((255, 255, 255))
pygame.draw.rect(screen, rectangle_color,
    (screen_width // 2 - new_width // 2,
    screen_height // 2 - new_height // 2,
    new_width, new_height))
pygame.display.flip()
```

III. ПРИМЕР РАБОТЫ КОДА



IV. ИЗМЕНЕНИЯ В КОДЕ

Для реализации плавного масштабирования вместо скачкообразного потребуется изменить подход к обработке коэффициента масштабирования во время обновления экрана. Вместо того чтобы изменять коэффициент масштабирования сразу на фиксированное значение при нажатии клавиш, можно использовать интерполяцию для плавного изменения масштаба. Необходимо добавить переменную для хранения желаемого масштаба и постепенно увеличивать или уменьшать текущий масштаб, чтобы достичь желаемого значения.

1. Переменные `currentscale` - текущее значение масштабирования, которое будет плавно изменяться и `targetscale` - желаемое значение масштабирования, которое будет устанавливаться по нажатиям клавиш.
2. Скорость масштабирования: `scalespeed` - определяет, насколько быстро будет происходить плавное изменение масштаба.
3. Логика изменения масштаба: после нажатия клавиши «вверх» и «вниз» `targetscale` обновляется, и постепенно приводит `currentscale` к `targetscale`, добавляя или вычитая `scalespeed`.

Ниже приведена обновленная часть кода с плавным масштабированием:

```
if current_scale < target_scale:
    current_scale += scale_speed
if current_scale > target_scale:
    current_scale = target_scale
elif current_scale > target_scale:
    current_scale -= scale_speed
if current_scale < target_scale:
    current_scale = target_scale
```

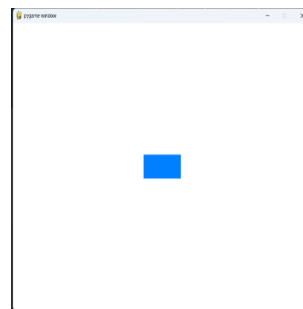
Эти изменения обеспечат плавный переход между размерами прямоугольника, улучшая пользовательский опыт.

Коршикова Дарья Валерьевна, магистрант кафедры информационных технологий автоматизированных систем БГУИР, korshikova@bsuirl.by.

Купчина Екатерина Валерьевна, магистрант кафедры информационных технологий автоматизированных систем БГУИР, e.kupchina@bsuirl.by.

Научный руководитель: Трофимович Алексей Фёдорович, старший преподаватель, заместитель декана ФИТУ, trofimaf@bsuir.by.

V. ПРИМЕР КОДА С ИЗМЕНЕНИЯМИ



VI. ВЫВОДЫ

Масштабирование графических элементов – важный аспект разработки пользовательских интерфейсов и визуализации данных. Понимание принципов аффинных преобразований и пропорциональности позволит разработчикам создавать более адаптивные и интуитивно понятные приложения. Приведенный выше пример демонстрирует реализацию масштабирования в Pygame, но принципы могут быть применены к другим графическим библиотекам и фреймворкам.

1. Васильев, Алексей Программирование на Python в примерах и задачах / А. Васильев – 2022. – С. 538-596.
2. Голд, Майкл Создание видеоигр с помощью PyGame / М. Голд – 2023. – С. 24-55.

АЛГОРИТМ ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ: ПРИНЦИПЫ И ПРИМЕРЫ

В данной статье будут рассмотрены основные методы восстановления изображений и приведем пример модификации кода на Python, использующего библиотеку OpenCV для выполнения этой задачи.

ВВЕДЕНИЕ

Восстановление изображений – это процесс улучшения качества изображений, которые были искажены или повреждены. Такие искажения могут быть вызваны шумом, размытием, недостаточной резкостью или даже потерей информации.

I. ПРИНЦИПЫ ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ

Существует несколько методов восстановления изображений, включая:

1. **Фильтрация:** фильтры (такие как гауссов фильтр или медианный фильтр) используются для удаления шума. Они работают на основе анализа соседних пикселей и исправляют значения пикселей на основании их окружения.
2. **Деконволюция:** этот метод позволяет восстановить оригинальное изображение, основываясь на модели процесса искажения (например, размытия). Часто используется для коррекции размытых изображений.
3. **Интерполяция:** применяется для восстановления потерянной информации в изображении, когда оно было уменьшено или обрезано. Методы интерполяции (линейная, кубическая) помогают достичь визуально приемлемых результатов.
4. **Преобразование Фурье:** позволяет анализировать частоты в изображении и восстанавливать утраченные детали, работая с частотной областью.

II. ПРИМЕР КОДА

Рассмотрим пример восстановления изображения с использованием медианного фильтра для удаления шума и функции `cv2.inpaint` для восстановления поврежденных областей. Для этого потребуется библиотека OpenCV.

```
import cv2
import numpy as np
```

Происходит загрузка изображения с помощью функции `cv2.imread`.

```
image = cv2.imread('input_image.jpg')
```

Добавление шума. Случайный шум добавляется к изображению для имитации искажения. Создание маски. Маска создается для областей изображения,

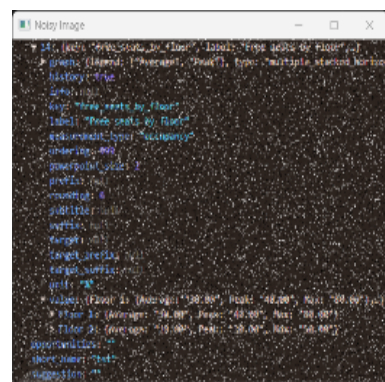
которые необходимо восстановить. В данном примере условие создания маски основано на значении одного из цветовых каналов. Происходит восстановление изображения с помощью фильтрации – применяется медианный фильтр для удаления шума и инпейнтинга – функция `cv2.inpaint` восстанавливает определенные области на изображении, основываясь на окружающих пикселях.

```
noise = np.random.randint
(0, 50, image.shape, dtype='uint8')
noisy_image = cv2.add(image, noise)
mask = np.zeros
(noisy_image.shape[:2], dtype='uint8')
mask[noisy_image[:, :, 0] > 200] = 255
denoised_image = cv2.medianBlur
(noisy_image, 5)
inpainted_image = cv2.inpaint
(denoised_image, mask, inpaintRadius=3,
flags=cv2.INPAINT_TELEA)
```

Отображение оригинального, зашумленного и восстановленного изображений

```
cv2.imshow('Original Image', image)
cv2.imshow('Noisy Image', noisy_image)
cv2.imshow('Inpainted Image',
inpainted_image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

III. ПРИМЕР РАБОТЫ КОДА



IV. ИЗМЕНЕНИЯ В КОДЕ

Чтобы улучшить представленный код необходимо внести несколько изменений, чтобы повысить качество результатов и сделать его более универсальным. Вот некоторые изменения и улучшения, которые можно внедрить:

1. Изменение типа шума. Вместо добавления случайного шума, можно использовать более распространенные типы шумов, такие как соль и перец. Этот тип шума более реалистичен и часто встречается в реальных изображениях.

```
def add_salt_and_pepper_noise(image,
    salt_prob, pepper_prob):
    noisy_image = np.copy(image)
    num_salt = np.ceil
    (salt_prob * total_pixels)
    coords = [np.random.randint(0, i - 1,
    int(num_salt)) for i in image.shape]
    noisy_image[coords[0], coords[1], :]
    = 255
    num_pepper = np.ceil
    (pepper_prob * total_pixels)
    coords = [np.random.randint(0, i - 1,
    int(num_pepper)) for i in image.shape]
    noisy_image[coords[0], coords[1], :]
    = 0
    return noisy_image
noisy_image = add_salt_and_pepper_noise
(image, salt_prob=0.02,
    pepper_prob=0.02)
```

2. Улучшение создания маски. Необходимо добавить размытие к маске, чтобы сделать края менее резкими и сгладить переходы. Для этого создается маска на основе цветов пикселей, которые представляют шум, что позволяет более точно отчитывать области, которые нужно восстановить.

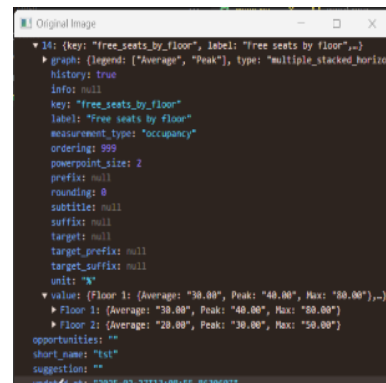
```
mask = np.zeros
(noisy_image.shape[:2],
    dtype='uint8')
mask[noisy_image[:, :, 0] == 255]
= 255
mask[noisy_image[:, :, 0] == 0]
= 255
```

3. Применение Гауссова размытия на маске. Это улучшит плавность распределения маски вокруг шумных пикселей, что сделает восстановление более естественным.

```
mask = cv2.GaussianBlur(mask,
    (5, 5), 0)
```

Эти улучшения помогут получить более качественное восстановление поврежденных областей в изображении.

V. ПРИМЕР РАБОТЫ КОДА ПОСЛЕ ВНЕСЕНИЙ ИЗМЕНЕНИЙ



VI. ВЫВОДЫ

Восстановление изображений – это важная область компьютерного зрения, используемая в различных приложениях, включая медицинскую визуализацию, фотографию и обработку видео. Методы, такие как фильтрация, деконволюция и инпейнтинг, помогают восстанавливать поврежденные изображения и улучшать их качество. Приведенный выше код демонстрирует способ удаления шума и восстановления части изображения с использованием Python и OpenCV. Эти принципы могут быть адаптированы и расширены для более сложных задач восстановления.

1. Васильев, А.. Программирование на Python в примерах и задачах / А. Васильев // 2020. – С. 16-69.
2. Брэдски, Г., Кэлер, А.. Изучаем OpenCV 3 / Г. Брэдски, А. Кэлер // 2017. – С. 81-101.

Купчина Екатерина Валерьевна, магистрант кафедры информационных технологий автоматизированных систем БГУИР, ekupchina@bsuir.by.

Коршикова Дарья Валерьевна, магистрант кафедры информационных технологий автоматизированных систем БГУИР, korshikova@bsuir.by.

Научный руководитель: Трофимович Алексей Фёдорович, старший преподаватель, заместитель декана ФИТУ, trofimaf@bsuir.by.

ДИНАМИЧЕСКОЕ РАЗРЕШЕНИЕ КОНФЛИКТОВ В ГЕТЕРОГЕННЫХ MAS НА ОСНОВЕ ТЕОРИИ ИГР С НЕПОЛНОЙ ИНФОРМАЦИЕЙ

Рассматривается подход к динамическому разрешению конфликтов в гетерогенных многоагентных системах на основе теории игр с неполной информацией. Предложена модель управления трафиком дронов в городской среде. Анализируются преимущества метода перед системами жёстких правил.

ВВЕДЕНИЕ

Рост использования дронов в городской среде требует новых подходов к управлению воздушным трафиком. Традиционные методы часто не учитывают динамичность таких систем. Теория игр с неполной информацией предлагает решение, позволяя агентам адаптироваться к изменениям. Разработка модели конфликтов на основе асимметричных игр для гетерогенных MAS является актуальной задачей.

I. МОДЕЛИРОВАНИЕ КОНФЛИКТОВ В АСИММЕТРИЧНЫХ ИГРАХ

Асимметричные игры отражают различия в стратегиях, ресурсах и информации между агентами, что характерно для гетерогенных MAS.

В отличие от классических симметричных моделей, асимметрия позволяет учесть уникальные параметры каждого участника, такие как приоритеты маршрутов, технические ограничения дронов и уровень доступа к данным.

Таблица 1 – Методы управления трафиком

Метод	Точность	Адаптивность	Сложность
Жёсткие правила	Низкая	Низкая	Низкая
Теория игр	Высокая	Высокая	Средняя
Машинное обучение	Средняя	Высокая	Высокая

Ключевым аспектом становится адаптивность стратегий: агенты корректируют свои действия на основе наблюдаемого поведения других участников и частичной информации о состоянии среды.

Для формализации конфликтов применяется аппарат Байесовых игр, где неполнота данных компенсируется вероятностными оценками намерений оппонентов. Это создаёт основу для прогнозирования конфликтных ситуаций и выбора оптимальных траекторий движения.

Михнюк Александр Иванович, студент ФИТиУ БГУИР, alexander.mikhniuk@gmail.com.

Савчиц Павел Сергеевич, студент ФИТиУ БГУИР, psavchits@gmail.com.

Научный руководитель: Хаджинова Наталья Владимировна, старший преподаватель кафедры ИТАС БГУИР, kha.jynova@bsuir.by.

II. ПРИМЕНЕНИЕ В УПРАВЛЕНИИ ТРАФИКОМ ДРОНОВ

В городских условиях дроны сталкиваются с динамическими препятствиями, включая здания, другие БПЛА и внезапные изменения маршрутов.

Предлагаемая модель интегрирует данные с датчиков и исторические паттерны движения для формирования адаптивных стратегий. Каждый агент оценивает риски столкновений, учитывая не только свои цели, но и вероятные действия соседних дронов, даже при отсутствии полной информации об их параметрах.

Алгоритмы на основе теории игр позволяют агентам пересчитывать маршруты в реальном времени, балансируя между эффективностью и безопасностью.

III. ВЫВОДЫ

Использование асимметричных игр с неполной информацией обеспечивает гибкость и адаптивность, критически важные для гетерогенных MAS. Результаты исследования подчёркивают потенциал теории игр в создании интеллектуальных систем, способных функционировать в условиях неопределённости. Дальнейшая работа будет направлена на интеграцию машинного обучения для улучшения прогнозирования поведения агентов и учёта дополнительных факторов, таких как погодные условия и правовые ограничения. Это открывает перспективы для безопасного масштабирования применения БПЛА в умных городах.

1. Ahmad F. Game theory applications in traffic management: A review of authority-based travel modelling: Travel Behaviour and Society / F. Ahmad. Netherlands, 2023.
2. Shamasoshoara A. Distributed Cooperative Spectrum Sharing in UAV Networks Using Multi-Agent Reinforcement Learning : arXiv / A. Shamasoshoara. USA, 2018.

СИСТЕМЫ УПРАВЛЕНИЯ РЕСУРСОЕМКИМИ КОНТЕЙНЕРИЗИРОВАННЫМИ ПРИЛОЖЕНИЯМИ

Рассматриваются системы управления ресурсоемкими контейнеризированными приложениями. Описываются возможности системы оркестрации контейнеризированными приложениями *Kubernetes*.

ВВЕДЕНИЕ

При решении задач развертывания ресурсоемких контейнеризированных приложений появляется необходимость в объединении ресурсов ЭВМ для образования рабочего кластера масштабируемых целевых контейнеров. Выбор систем управления контейнеризированными приложениями определяет эффективность и целесообразность разворачиваемой системы. Ограниченность вычислительных ресурсов требует использование наиболее современных систем оркестрации.

I. АКТУАЛЬНЫЕ ВОЗМОЖНОСТИ СИСТЕМ ОРКЕСТРАЦИИ

Развитие систем оркестрации обусловлено необходимостью в централизованном управлении ограниченными вычислительными ресурсами. Тенденции к снижению материальных затрат на развертывание приложений обуславливают низкоуровневый интерфейс общения модулей систем управления кластеров контейнеров.

Стремительный рост сетевых технологий обусловил необходимость реализации сетевых интерфейсов общения разворачиваемых контейнеризированных приложений. Комбинация и доработка существующих сетевых решений позволила реализовать эффективные механизмы распределения сетевого трафика между элементами масштабируемых ресурсоемких контейнеризированных приложений.

Низкоуровневый характер модулей управления контейнеризации приложений позволил реализовать автоматический контроль масштабирования рабочих модулей приложений, позволяя оптимизировать мониторинг реализуемой программной системы.

Фокусировка на управлении контейнеризацией, по отношению к виртуализации, обусловила эффективное использование как памяти, так и вычислительных ресурсов [1].

Ряд систем управления ресурсоемкими контейнеризированными приложениями учитывают и реализуют совместимость с провайдерами облачных технологий.

II. СРАВНЕНИЕ СИСТЕМ ОРКЕСТРАЦИИ

Рынок программных продуктов предлагает широкий спектр систем управления контейнеризированными приложениями. Большинство систем управления контейнеризированными приложениями предоставляет услуги на основе лицензий *Open Source* (табл.1.):

Таблица 1 – Сравнение лицензирования и эксплуатации систем оркестрации

Система управления	Тип лицензии
<i>Kubernetes</i>	<i>Open Source</i>
<i>Docker Swarm</i>	<i>Open Source</i>
<i>Apache Mesos</i>	<i>Open Source</i>
<i>OpenShift</i>	Коммерческая
<i>Nomad</i>	<i>Open Source</i>
<i>Rancher</i>	<i>Open Source</i>
<i>Amazon ECS</i>	Проприетарная

После анализа возможностей систем управления ресурсоемкими контейнеризированными приложениями имеем следующие индексы управления ресурсоемкими приложениями (ИУРП) (табл.2.):

Таблица 2 – Индекс управления ресурсоемкими приложениями

Система управления	ИУРП
<i>Kubernetes</i>	9.5
<i>Docker Swarm</i>	6.4
<i>Apache Mesos</i>	9.0
<i>OpenShift</i>	9.8
<i>Nomad</i>	8.5
<i>Rancher</i>	8.0
<i>Amazon ECS</i>	7.6

Назначение ИУРП происходило по следующей формуле:

$$IURP = \frac{C_i}{C_{max}} \quad (1)$$

где *IURP* – индекс управления ресурсоемкими приложениями для конкретной системы; C_i – средняя геометрическая оценка *i*-ой системы по критериям, определяемым экспертами; C_{max} максимальная оценка среди всех рассматриваемых систем. При назначении индексов использовались критерии:

- управления ресурсами *CPU/GPU*;
- распределения ресурсоемких задач;
- ограничения ресурсов контейнера;
- оптимизации сходства (*affinity*);
- управления памятью для нагрузок;
- оптимизации планирования вычислений;
- поддержки хранилищ данных;
- автомасштабирования рабочих нагрузок;
- приоритизации приложений с состояниями (*stateful*);
- приоритизации критических задач;
- эффективности больших кластеров.

Наилучшим результатом обладает система *OpenShift*, построенная на базе *Kubernetes*. *OpenShift*

расширяет функциональность *Kubernetes*, предлагая дополнительные инструменты для управления, безопасности и автоматизации. Наихудшим результатом обладает *Docker Swarm*, ввиду ограниченных возможностей в управлении ресурсами и масштабируемости, что делает систему менее эффективной для обработки сложных задач.

Рассмотрение оркестратора *Kubernetes* покрывает фундаментальную основу возможностей сравниваемых систем, в том числе *OpenShift*.

III. СИСТЕМА ОРКЕСТРАЦИИ РЕСУРСОЕМКИМИ КОНТЕЙНЕРИЗИРОВАННЫМИ ПРИЛОЖЕНИЯМИ *Kubernetes*

Глобально кластеры *Kubernetes* строятся из мастер-узлов и рабочих узлов. Архитектура мастер-узлов *Kubernetes* отражает эффективность разделения ответственности между компонентами управления и исполнения (см. рис.1.)

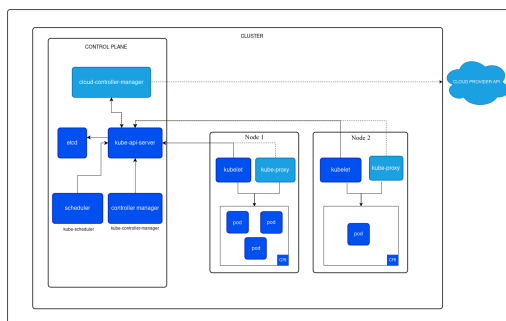


Рис. 1 – Структура управляющего узла *Kubernetes*

Централизация функций управления на мастер-узлах (*API*-сервер, *etcd*, планировщик) при делегировании исполнения контейнеров рабочим узлам образует структурную целостность системы масштабирования вычислительных ресурсов кластера.

Сетевая архитектуры *Kubernetes* демонстрирует наличие гибких механизмов взаимодействия через стандартизированные интерфейсы *CNI*, обеспечивающие прозрачную коммуникацию между компонентами распределенных приложений независимо от топологии инфраструктуры.

Применение *Gateway API*, расширяет функциональность управления сетевым трафиком. *Gateway API* предоставляет детализированное управление маршрутами и слежением за состоянием сервисов.

Исследование возможностей работы *Kubernetes* с системами управления базами данных

выявляет преимущества механизма операторов, реализующего высокоуровневую абстракцию над процессами администрирования *PostgreSQL*, *MongoDB*, *MySQL*. Внедрение *StatefulSet*-контроллеров обеспечивает детерминированное поведение приложений с сохранением состояния в условиях динамических изменений инфраструктуры.

Анализ механизмов масштабирования *Kubernetes* показывает эффективность комбинированного применения *Horizontal Pod Autoscaler* и *Vertical Pod Autoscaler* для оптимизации использования вычислительных ресурсов кластера.

Системы управления ресурсами *Kubernetes* реализует дифференцированный подход к распределению вычислительных мощностей между приложениями различных классов. Внедрение метрик *request* и *limit* в сочетании с механизмами *Quality of Service* создает предсказуемую среду выполнения для ресурсоемких приложений даже в условиях конкуренции за ресурсы кластера.

Оркестратор *Kubernetes* демонстрирует открытость архитектуры для взаимодействия с внешними системами мониторинга, логирования, управления конфигурациями.

Интегрируемость с ресурсами *ArgoCD* и *Jenkins* позволяет использовать методологию *GitOps*, упрощая поставку приложений на пользовательские сервера [2].

IV. Выводы

Исследовательская работа предоставляет данные необходимые для выбора системы управления ресурсоемкими контейнеризированными приложениями. В массиве актуальных решений в области оркестрации компьютеризацией выявлены лидирующие системы. Описаны основные возможности системы оркестрации *Kubernetes*, обоснованной фундаментальным характером системной архитектуры, применяемой в инструменте. Концентрация разработчиков на актуализации возможностей системы *Kubernetes* позволила использовать ее для разработки наиболее актуальных дочерних решений – *OpenShift*.

1. Эдриен, М. Использование Docker / М. Эдриен, А. В. Снастина, А. А. Маркелов. // М.: ДМК Пресс – 2017. – №2 – С. 5-41.
2. Ваяс, Дж. Kubernetes изнутри / Дж. Ваяс, К. Лав // М.: ДМК Пресс – 2022. – №1. – С. 32-57.

Проход Андрей Олегович, студент кафедры информационных технологий автоматизированных систем БГУИР, prokhorandrey03@gmail.com.

Ярмолик Валерий Иванович, старший преподаватель информационных технологий автоматизированных систем БГУИР, v.jarmolik@bsuir.by

СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ СЛОВ В ВЕКТОРНЫХ БАЗАХ ДАННЫХ

Рассматриваются методы векторного представления слов в векторных базах данных. Исследуются подходы к созданию и обработке векторных представлений, включая прогнозные (Word2Vec), контекстуальные (BERT) и графовые модели (GNN), а также классический латентно-семантический анализ (LSA) и глобальные векторные представления (GloVe). Основное внимание уделяется способности методов справляться с разнообразием задач обработки текстовой информации.

ВВЕДЕНИЕ

Векторные базы данных становятся ключевым инструментом для эффективного поиска и обработки семантически связанных данных, а выбор методов векторного представления слов (word embeddings) напрямую влияет на их производительность и точность. В последние годы появилось множество подходов — от классических (Word2Vec, GloVe) до современных контекстных моделей (BERT), — каждый из которых обладает своими преимуществами и ограничениями в задачах индексирования, поиска и кластеризации. Однако при интеграции в векторные базы данных эти методы демонстрируют разную эффективность в зависимости от масштаба данных, требований к скорости и точности. Исследование ключевых методов векторного представления, оценивание их применимости в реальных сценариях работы с векторными базами, и выявление оптимальных стратегий выбора эмбедингов для различных задач способствует разработке более эффективных и масштабируемых систем обработки информации, а также повышает точность поиска и анализа данных.

I. ВЕКТОРНОЕ ПРЕДСТАВЛЕНИЕ СЛОВ

Векторное представление слов (word embeddings) — это способ преобразования текстовой информации в многомерные числовые векторы фиксированной размерности, которые сохраняют семантические и синтаксические связи между словами [1]. Такие представления позволяют учитывать смысловые связи между словами, их контекст и семантику, делая возможным применение сложных алгоритмов машинного обучения для обработки и анализа текстов. Современные методы, от классических до контекстно-зависимых, различаются по способу кодирования информации, что влияет на их применимость в различных задачах — поиске, классификации или кластеризации текстов.

II. ТРЕБОВАНИЯ К МЕТОДАМ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ СЛОВ В ВЕКТОРНЫХ БАЗАХ ДАННЫХ

При выборе подхода для интеграции в векторные базы данных ключевыми становятся несколько критериев: способность сохранять семантические и синтаксические отношения между лексическими единицами в компактном векторном пространстве, масштабируемость для обработки больших объемов текстовых данных, эффективность индексирования и поиска в многомерных пространствах, адаптивность к различным предметным областям и языковым особенностям, устойчивость к полисемии и другим лингвистическим явлениям, баланс между размерностью векторов и вычислительной сложностью операций, а также поддержка динамического обновления представлений без полного переобучения модели.

III. СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ СЛОВ

Сравнительный анализ методов векторного представления слов, таких как LSA, Word2Vec, GloVe, BERT и графовые нейронные сети, позволяет выявить их применимость в задачах, связанных с векторными базами данных, где LSA подходит для выявления скрытых семантических связей, Word2Vec и GloVe эффективны для семантического поиска, BERT учитывает контекст, а графовые модели — структурные связи. Выбор метода, определяемый задачей, требованиями к точности, скорости и доступным ресурсам, критически важен для эффективной работы векторных баз данных.

1. Lychenko, N. Comparison of the effectiveness of word vector representation methods for determining the sentiment of texts / N. Lychenko, A. Sorokova // University of Central Florida – 2019. – Vol. 6. – P. 1-6.

Савоневская Маргарита Олеговна, студентка ФИТиУ БГУИР, go.to.mychannel745@gmail.com.

Столярова Виолетта Витальевна, студентка ФИТиУ БГУИР, violettastolarova@icloud.com.

Черникова Лолита Александровна, студентка ФИТиУ БГУИР, lolita190511@gmail.com.

Научный руководитель: Трофимович Алексей Фёдорович, старший преподаватель кафедры ИТАС БГУИР, trofimaf@bsuir.by.

ИСПОЛЬЗОВАНИЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ (RL) ДЛЯ ПРЕДОТВРАЩЕНИЯ ТУПИКОВЫХ СИТУАЦИЙ В СИСТЕМАХ С ОБЩИМИ РЕСУРСАМИ

Рассматривается применение обучения с подкреплением (RL) для предотвращения тупиковых ситуаций (deadlock) в системах с общими ресурсами. Предлагается использование Q-обучения в условиях ограниченной наблюдаемости среды, где агенты адаптивно учатся избегать коллизий при доступе к ресурсам.

ВВЕДЕНИЕ

Возникновение тупиковых ситуаций в системах с общими ресурсами представляет собой серьезную проблему, которая препятствует их эффективному функционированию. Применение методов обучения с подкреплением (Reinforcement Learning, RL), основным из которых является Q-обучение, позволяет разрабатывать стратегии предотвращения коллизий даже в условиях ограниченной наблюдаемости среды.

I. ТУПИКОВЫЕ СИТУАЦИИ В СИСТЕМАХ С ОБЩИМИ РЕСУРСАМИ

Тупиковые ситуации (deadlocks) — это критические состояния, при которых группа процессов взаимно блокируется из-за циклического ожидания недоступных ресурсов, удерживаемых другими процессами [1]. В распределённых системах проблема усугубляется неполной наблюдаемостью и децентрализованным управлением, что требует адаптивных методов (например, Reinforcement Learning).

II. ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ (RL) КАК ПОДХОД ДЛЯ ПРЕДОТВРАЩЕНИЯ ТУПИКОВ

Обучение с подкреплением (Reinforcement Learning, RL) представляет собой парадигму машинного обучения, в которой агент обучается принимать оптимальные решения через взаимодействие со средой. RL использует систему вознаграждений и штрафов, позволяя агенту самостоятельно находить стратегии, максимизирующие долгосрочную выгоду. Ключевое преимущество RL в данном контексте — способность адаптироваться к изменяющимся условиям без явного перепрограммирования.

III. ОСОБЕННОСТИ ИЗБЕГАНИЯ КОЛЛИЗИЙ ЧЕРЕЗ Q-ОБУЧЕНИЕ

Основная задача алгоритма Q-обучения — сформировать оптимальную стратегию поведения,

обновляя значения Q-функции на основе опыта, полученного в процессе взаимодействия со средой. Процесс Q-обучения основан на поэтапном пересмотре значений Q-функции. Через итеративное обновление Q-таблицы алгоритм выявляет паттерны, ведущие к deadlock, и вырабатывает стратегию их избегания. В работе он использует только локальную информацию, адаптируясь к изменениям системы без полного перерасчета политики. Каждый раз, когда программа совершает действие и получает обратную связь от среды (в виде награды и перехода в новое состояние), он корректирует свои представления о ценности этого действия [2]. Этот итеративный подход позволяет со временем всё точнее определять, какие решения будут наиболее выгодными в долгосрочной перспективе, при этом избегая тупиковые ситуации.

IV. КОНЦЕПТУАЛЬНАЯ МОДЕЛЬ ИНТЕГРАЦИИ Q-ОБУЧЕНИЯ В СИСТЕМЫ С ОБЩИМИ РЕСУРСАМИ

Предлагается концептуальная модель, в которой агенты, использующие Q-обучение, взаимодействуют с системой с общими ресурсами, получая частичную информацию о ее состоянии и принимая решения, направленные на предотвращение коллизий и тупиков. Модель включает следующие компоненты: агенты (программные сущности, обучающиеся с использованием Q-обучения для принятия решений), среда (все элементы системы, с которыми взаимодействуют агенты), награды (определяются на основе эффективности работы системы, включая показатели предотвращения тупиков).

1. Chen, M. Deadlock-Detection via Reinforcement Learning / M. Chen, L. Rabelo // University of Central Florida – 2017. – Vol. 6. – P. 1-6.
2. Watkins, C. Q-Learning / C. H. Watkins, P. Dayan // Q-Learning – 1992. – P. 281-285.

Савоневская Маргарита Олеговна, студентка ФИТиУ БГУИР, go.to.mychannel745@gmail.com.

Столярова Виолетта Витальевна, студентка ФИТиУ БГУИР, violettastolarova@icloud.com.

Черникова Лолита Александровна, студентка ФИТиУ БГУИР, lolita190511@gmail.com.

Научный руководитель: Хаджинова Наталья Владимировна, старший преподаватель кафедры ИТАС БГУИР, kha.jynova@bsuir.by.

СИСТЕМА АВТОМАТИЧЕСКОГО РАСПОЗНАВАНИЯ ДОРОЖНЫХ ЗНАКОВ

В данной работе представлена система автоматического распознавания дорожных знаков на основе модели YOLO (You Only Look Once). Описаны этапы сбора и предобработки данных, анализ дисбаланса классов и методы балансировки, а также проведена оценка эффективности с помощью кросс-валидации и стандартных метрик. Экспериментальные исследования показали, что применение комбинации андерсемплинга и аугментации позволяет достичь средней точности (mAP) равной 0.570.

ВВЕДЕНИЕ

Создание автономных транспортных систем требует высокой точности распознавания дорожных знаков для обеспечения безопасности. Актуальность работы обусловлена необходимостью интеграции современных методов компьютерного зрения в реальные условия. Автопилот должен не только определять и интерпретировать знаки, но и делать это стабильно при изменениях освещённости, погодных условиях, загрязнении и других помехах.

I. ОБРАБОТКА ДАННЫХ И ВЫБОР МОДЕЛИ

Для реализации системы использовались два набора изображений: один содержит 104 тыс., другой — 180 тыс. изображений, адаптированных под белорусские дорожные знаки. Предобработка включала нормализацию размеров, андерсемплинг для переизбыточных классов и аугментацию (повороты, освещение, масштаб) для малочисленных. Такой подход позволил выровнять распределение классов и повысить устойчивость модели к условиям реального применения.

Выбор YOLO обусловлен её высокой скоростью (до 45 FPS) и точностью (mAP = 0.570). Модель локализует и классифицирует объекты за один проход, что делает её подходящей для систем реального времени. Для оценки эффективности использовались кросс-валидация и метрики (precision, recall, F1-score). Также проведены стресс-тесты модели в условиях дождя, снега и ночного времени.

II. ПОВЫШЕНИЕ ТОЧНОСТИ РАСПОЗНАВАНИЯ

Анализ экспериментальных данных показал, что основные трудности возникают при распознавании малочисленных классов знаков, что негативно влияет на общие показатели precision. Это связано с тем, что модель не получает достаточно примеров для обучения на редких классах. Для снижения этого эффекта использовалась комбинация техник:

синтетическая генерация изображений, взвешенные функции потерь и тонкая настройка гиперпараметров.

Для визуализации эффективности работы модели в условиях дисбаланса была построена PR-кривая, демонстрирующая зависимость precision и recall при варьировании порога принятия решения (см. рис.1).

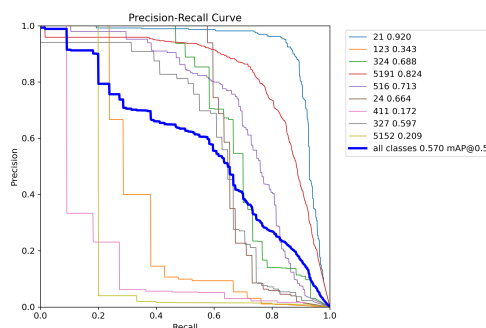


Рис. 1 – PR-кривая для оптимизированной модели YOLO

III. ВЫВОДЫ

Применение аугментации и андерсемплинга позволяет повысить точность распознавания дорожных знаков за счёт устранения дисбаланса данных. Разработанная система демонстрирует удовлетворительную точность (mAP = 0.570), что подтверждено PR-кривой. Отдельные классы, такие как «Стоп» и «Уступите дорогу», распознаются с высокой точностью, а редкие знаки требуют доработки.

1. Redmon J. et al. YOLOv3: An Incremental Improvement // arXiv, 2018.
2. Гудфеллоу Я. Глубокое обучение. М.: ДМК Пресс, 2022.
3. Krizhevsky A. ImageNet Classification with Deep Convolutional Neural Networks // NIPS, 2012.

Сморищок Кирилл Дмитриевич, студент кафедры информационных технологий автоматизированных систем БГУИР, s.kirill150282@mail.ru.

Хмыз Дарья Дмитриевна, магистрант кафедры информационных технологий автоматизированных систем БГУИР, shilinadarya@gmail.com.

Научный руководитель: Шилин Леонид Юрьевич, декан факультета информационных технологий и управления БГУИР, доктор технических наук, профессор, dekfitu@bsuir.by.

ПОСТРОЕНИЕ ДЕРЕВА КРАТЧАЙШИХ РАССТОЯНИЙ МИНИМАЛЬНОЙ СТОИМОСТИ

В статье рассматриваются методы скаляризации (свёртки) весов для задачи построения дерева кратчайших расстояний минимальной стоимости.

ВВЕДЕНИЕ

В геоинформационных системах часто возникает задача маршрутизации по нескольким критериям. Например, в транспортной сети с векторными весами дуг, представляющими расстояние и стоимость, необходимо найти компромиссное решение, позволяющее эффективно учитывать оба этих критерия. Известный алгоритм построения максимального потока минимальной стоимости [1] не подходит для нашего случая. Перспективным является поиск компромиссного решения через скаляризацию (свёртку) векторных весов дуг транспортной сети.

I. ПОСТАНОВКА ЗАДАЧИ

Предположим, что в транспортной сети каждое ребро (дуга) имеет два веса: d_{ij} - расстояние в километрах и c_{ij} - стоимость в денежных единицах. Понятно, что дерево кратчайших расстояний и дерево маршрутов минимальной стоимости в общем случае не совпадут. Как же получить компромиссное решение? Ключевым моментом в получении компромиссного решения является способ свёртки векторных весов дуг; этих способов достаточно много и все они теоретически базируются на подходе, связанном с весовыми коэффициентами. Рассмотрим два основных подхода к свёртке весов: линейная комбинация весов и нормализация весов [2].

II. ЛИНЕЙНАЯ КОМБИНАЦИЯ ВЕСОВ

Этот метод заключается в линейной комбинации двух весов (расстояния d_{ij} и стоимости c_{ij}) с использованием коэффициента важности (полезности) λ (где $0 \leq \lambda \leq 1$):

$$w_{ij} = \lambda d_{ij} + (1 - \lambda) c_{ij} = \lambda_1 d_{ij} + \lambda_2 c_{ij}. \quad (1)$$

Если λ (λ_1) ближе к 1, то при работе алгоритма больший вес приобретает расстояние; в противном случае (λ ближе к 0) - стоимость проезда (λ_2), т.е. выбор зависит от приоритетов критериев. К этому методу близок метод взвешенного суммирования для сбалансированности разных критериев, например, стоимости:

$$w_{ij} = \lambda d_{ij} + (1 - \lambda)(1 - c_{min}/c_{ij}), \quad (2)$$

где c_{min} - минимальная стоимость среди всех стоимостных весов дуг, что позволяет добавить ранжирование по стоимости - чем больше стоимость, тем больший «довесок» она вносит в скалярный вес дуги. Недостатком этих методов является то, что у нас «суммируются» либо веса с различными единицами измерения (км и ден. ед.), либо к одной единице измерения добавляется безразмерная величина.

III. НОРМАЛИЗАЦИЯ ВЕСОВ

В этом методе сначала производится нормализация обоих весов к единому диапазону (например, от 0 до 1), а затем их комбинация с весовыми коэффициентами:

$$d'_{ij} = \frac{d_{ij} - d_{min}}{d_{max} - d_{min}}, c'_{ij} = \frac{c_{ij} - c_{min}}{c_{max} - c_{min}}, \quad (3)$$

$$w_{ij} = \lambda d'_{ij} + (1 - \lambda) c'_{ij} = \lambda_1 d'_{ij} + \lambda_2 c'_{ij}, \quad (4)$$

где d_{min}, d_{max} - минимальное и максимальное расстояния, а c_{min}, c_{max} - минимальная и максимальная стоимости среди всех дуг сети.

Этот метод более предпочтителен, так как веса приводятся к единой безразмерной шкале, что делает результаты более объективными. Выбор значения $\lambda()$.

ЗАКЛЮЧЕНИЕ

Таким образом, в графе транспортной сети векторные веса рёбер заменяются на скалярные веса $w_{ij}, , .(, *, ^\circ), [3]$.

СПИСОК ЛИТЕРАТУРЫ

- [1] Ху, Т. Целочисленное программирование и потоки в сетях. - М.: Мир, 1974. - С. 212-215.
- [2] Системный анализ и принятие решений в проектной и управленческой деятельности : учеб.-метод. пособие / Б. В. Никульшин [и др.]. - Минск : БГУИР, 2021. - 72 с.
- [3] Coley H.W., Moon J.D. Shortest paths in networks with vector weights. // J. Optim. Theory and Appl., 1985, v.46, № 1, p.79-86.

Служалик Владимир Юльевич, магистрант кафедры информационных технологий автоматизированных систем БГУИР, sluzhalik@bsuir.by.

Научный руководитель: Севернёв Александр Михайлович, кандидат технических наук, доцент, severnev@bsuir.by.

АЛГОРИТМ РАСЧЕТА ДОХОДНОСТИ ВАЛЮТНОГО ПОРТФЕЛЯ

Рассматривается алгоритм расчета доходности валютного в условиях недостатка данных по конвертациям курсов валют.

ВВЕДЕНИЕ

Для расчета доходности валютного портфеля пользователю необходимо создать портфель, используя форму в веб-приложении. При создании портфеля пользователь может добавить два типа валют: обычные валюты (например, американский доллар, российский рубль, польский злотый) и различные виды криптовалют. Для каждой валюты пользователь должен указать её количество в процентах от общего объема портфеля. Все валюты в совокупности должны составлять 100 %. Например, пользователь может создать следующий портфель, включающий как обычные валюты, так и криптовалюты: 20 % USD, 15 % PLN, 11 % AED, 24 % BTCUSDT, 30 % ETHUSDT. При создании портфолио пользователю необходимо указать базовую валюту, по отношению к которой будет рассчитываться доходность. Система предоставляет широкий выбор валют. В связи с невозможностью хранения прямых курсов обмена валют, например, в случае отсутствия информации о курсе целевой валюты по отношению к базовой на определенную дату, система использует специальные алгоритмы для расчета относительных курсов валют относительно базовой.

I. АЛГОРИТМ НАХОЖДЕНИЯ КУРСА ПРОИЗВОЛЬНОЙ ВАЛЮТЫ ПО ОТНОШЕНИЮ К ВЫБРАННОЙ

Задается диапазон дат для получения исторических данных, базовая валюта, которая будет конвертироваться в целевую валюту, и целевая валюта. Результатом работы алгоритма является список дат и курсов конвертации базовой валюты в целевую. На первом этапе производится поиск в истории курсов на указанную дату информации о прямой конвертации валют. Если результат найден, возвращается полученное значение. В случае отсутствия данных производится переход ко второму шагу. На втором шаге производится зеркальный поиск курса: целевой валюты к базовой. Если такой курс найден, возвращается обратное значение, которое считается курсом конвертации базовой валюты в целевую. При отсутствии данных производится переход к третьему шагу. На третьем шаге производится относительный поиск [1]. В системе имеются валюты, которые гарантированно конвертируются как в базовую валюту, так и в целевую. Для этой конвертации используется российский рубль (в качестве промежуточной валюты) [2]. Находится курс конвертации промежуточной валюты в базовую и курс конвертации промежуточной валюты в целе-

вую. Рассчитывается отношение курса конвертации промежуточной валюты в целевую к курсу конвертации промежуточной валюты в базовую. Полученное отношение считается курсом конвертации базовой валюты в целевую. Алгоритм обеспечивает высокую степень надежности за счет использования нескольких уровней поиска, что позволяет минимизировать количество пропущенных данных. Такой подход гарантирует корректность и полноту предоставляемой информации даже при отсутствии прямых курсов в исходных источниках. Алгоритм расчета курса произвольной валюты к заданной приведен на рисунке 1.

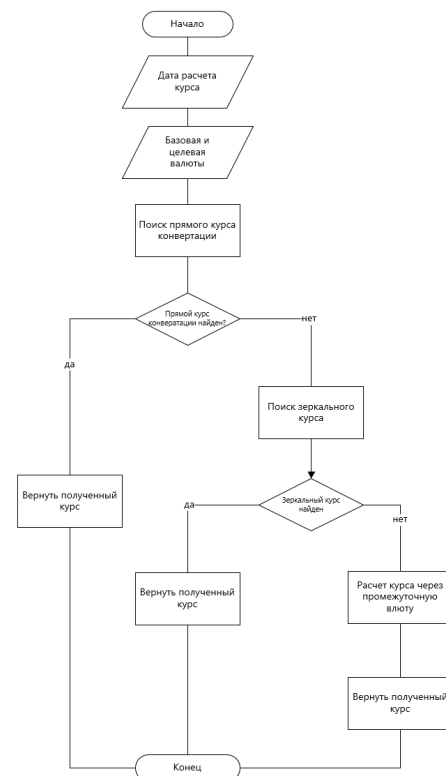


Рис. 1 – Блок-схема алгоритма расчета курса валюты на определенную дату

II. АЛГОРИТМ РАСЧЕТА ДОХОДНОСТИ ПОРТФЕЛЯ

Алгоритм расчета доходности портфеля приведен на рисунке 2.



Рис. 2 – Блок-схема алгоритма расчета доходности портфеля

Для расчета доходности портфеля валютит пользователь указывает временной период, за который необходимо рассчитать доходность. Далее находится история курсов всех классических валют по отношению к базовой валюте, указанной при создании портфеля. Дополнительно находится история курсов доллара по отношению к базовой валюте портфеля, так как доллар используется для конвертации курса криптовалюты в базовую валюту, так как система хранит курсы криптовалют по отношению к доллару. Используя найденную историю курсов доллара курс криптовалюты конвертируется в курс базовой валюты. Далее находятся курсы всех валют, включенных в портфолио, на начало указанного периода. На следующем этапе для каждой даты в указанном диапазоне для каждой валюты (как классической, так и криптовалюты) рассчитывается следующее значение: отношение разности курса валюты на указанную дату и курса на начало периода к курсу на начало указанного периода. Это значение считается доходностью валюты на указанную дату. Таким образом, для каждой даты из диапазона на рассчитывается доходность по формуле:

$$\frac{C_i - C_0}{C_0} \quad (1)$$

В указанной формуле C_0 - курс валюты на начало целевого периода, C_i - курс на целевую дату расчета. На последнем этапе для каждой из дат диапазона рассчитывается следующее значение: сумма произведений доходности каждой из валют на дату и процента содержания валюты в портфеле. Полученное значение считается доходностью всего портфеля на указанную дату. В случае, если в состав портфеля включена базовая валюта, указанная при его создании, её доходность на протяжении всего анализируемого временного периода принимается равной нулю, поскольку она используется в качестве эталонной единицы измерения стоимости других валют и, соответственно, не подлежит переоценке. Таким образом, результатом работы алгоритма является два массива: массив дат, сформированный на основе указанного периода, и массив доходности в процентах для каждой из дат диапазона.

III. Выводы

Алгоритм расчета доходности валютного портфеля представляет собой эффективный инструмент оценки доходности валютных активов. Преимуществом разработанного алгоритма является использование нескольких уровней поиска и конвертации курсов валют, что позволяет работать с неполными данными и корректно учитывать динамику как традиционных валют, так и криптовалют. Использование таких методов, как зеркальный и относительный поиск, делает алгоритм устойчивым к недостатку информации о прямых курсах валют. Алгоритм позволяет проводить детализированный расчет доходности для каждой валюты, что дает пользователю полное представление о динамике портфеля. Кроме того, расчеты производятся с учетом структуры портфеля и доли каждой валюты, что обеспечивает точную оценку совокупной доходности. Подобный подход делает алгоритм универсальным инструментом как для анализа исторических данных, так и для поддержки инвестиционных решений.

1. Сарно, Л., Тейлор, М. П. Экономика обменных курсов / Л. Сарно, М. П. Тейлор // Издательство: Cambridge University Press, 2003. – 327 с.
2. Центральный банк РФ [Электронный ресурс] / Нац. центр финанс информ. Российской Федерации – Москва, 2025. – Режим доступа: <https://www.cbr.ru/>. – Дата доступа: 07.04.2025.

Столбанов Артем Александрович, студент кафедры информационных технологий автоматизированных систем, artsiomst2004@gmail.com.

Ярмолик Валерий Иванович, старший преподаватель кафедры информационных технологий автоматизированных систем, v.jarmolik@bsuir.by.

АЛГОРИТМ ВЗАИМОДЕЙСТВИЯ КОМПОНЕНТОВ НА СТОРОНЕ КЛИЕНТА

В статье рассматривается алгоритм взаимодействия компонентов на стороне клиента в условиях динамически обновляемого интерфейса. Описываются этапы инициализации, синхронизации состояний и оптимизации рендеринга. Предложенный алгоритм может быть применён в архитектурах типа MVVM и SPA. Приводится формализация логики передачи данных между компонентами.

ВВЕДЕНИЕ

Современные клиентские приложения строятся по компонентному принципу. Это означает, что интерфейс разбит на независимые логические блоки — компоненты, которые взаимодействуют между собой. Эффективность такого взаимодействия определяет производительность и стабильность приложения.

Типовая проблема — избыточные перерендеры или неконсистентное состояние компонентов. Поэтому требуется алгоритм, минимизирующий количество событий синхронизации и обеспечивающий предсказуемость передачи данных.

ОПИСАНИЕ АЛГОРИТМА

Пусть C_i — компонент с локальным состоянием $S_i(t)$ в момент времени t . Пусть $D_{i,j}$ — данные, передаваемые от компонента C_i к C_j .

Обозначим событие изменения состояния как δS_i , и событие передачи данных как $\tau_{i,j}$.

Алгоритм взаимодействия включает следующие шаги:

1. Инициализация всех C_i , установка начального $S_i(0)$;
2. Обнаружение изменений состояния: если $\delta S_i \neq 0$, создаётся очередь обновлений;
3. Уведомление подписчиков:

$$\tau_{i,j}(t) = f(S_i(t))$$

где f — функция трансформации состояния в данные;

4. Обновление состояний получателей по правилу:

$$S_j(t+1) = g(S_j(t), \tau_{i,j}(t))$$

где g — функция применения входящих данных;

5. Оптимизация рендеринга: перерендер происходит только если

$$S_j(t+1) \neq S_j(t)$$

Таким образом, алгоритм предполагает поэтапную обработку изменений состояния компонентов и передачу этих изменений только тем частям интерфейса, которым они действительно необходимы.

ПРЕИМУЩЕСТВА

Разработанный алгоритм взаимодействия компонентов на стороне клиента обладает рядом существенных преимуществ. Прежде всего, он обеспечивает предсказуемость процессов синхронизации между компонентами, что особенно важно в условиях динамически изменяющегося интерфейса. Благодаря изоляции локальных состояний каждого компонента и чёткому управлению их изменениями, удаётся избежать нежелательных побочных эффектов и конфликтов данных.

Кроме того, алгоритм минимизирует количество ненужных перерендеров, что положительно сказывается на производительности клиентского приложения и его отклике на действия пользователя. Это достигается за счёт встроенной логики сравнения состояний и передачи только действительно изменённых данных.

ЗАКЛЮЧЕНИЕ

Алгоритм взаимодействия клиентских компонентов с учётом изменения состояния и оптимизации обновлений позволяет строить более устойчивые и производительные интерфейсы. В условиях увеличивающейся сложности SPA-приложений этот подход может стать основой для архитектурных решений.

ЛИТЕРАТУРА

1. Грубер И.Ю. Современные JavaScript-фреймворки. — Минск: ITBooks, 2022.
2. Dan Abramov. *A Complete Guide to useEffect*. — React Blog, 2021.
3. Preact Team. Signals: Fine-Grained Reactivity for the Modern Web. — 2023.

Струкова Алина Александровна, магистрант кафедры информационных технологий автоматизированных систем БГУИР, alinastrukovaa@gmail.com.

Научный руководитель: Шилин Леонид Юрьевич, доктор технических наук, профессор, декан факультета информационных технологий БГУИР, dekfitu@bsuir.by.

ИСПОЛЬЗОВАНИЕ BASH СКРИПТОВ ДЛЯ РАБОТЫ С ОПЕРАЦИОННОЙ СИСТЕМОЙ WINDOWS

В статье рассматривается использование Bash-скриптов для работы с операционной системой Windows, что открывает новые возможности для автоматизации задач и управления системой.

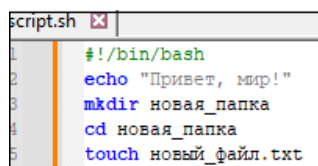
ВВЕДЕНИЕ

Bash (Bourne Again SHell) – интерпретатор командной строки, используемый в ОС Linux и Unix. Он также доступен в Windows 10 Anniversary Update и более поздних версиях. Хотя Windows использует Windows PowerShell в качестве своего основного командного интерпретатора, Bash-скрипты все равно могут быть полезными для работы с Windows: для автоматизации выполнения рутинных задач, выполнения сложных действий, которые не предусмотрены стандартными средствами Windows, для интеграции с другими инструментами и сценариями, для использования в качестве утилит, для использования при разработке и тестировании программного обеспечения под Windows. Bash-скрипты обычно имеют более простой и понятный синтаксис по сравнению с cmd или PowerShell [1].

I. УСТАНОВКА И ИСПОЛЬЗОВАНИЕ BASH НА WINDOWS

Bash-скрипты могут быть запущены на различных ОС, включая Windows с помощью Windows Subsystem for Linux (WSL). Установка WSL и Bash на Windows 10/11: установить Bash, активировав функцию "Windows Subsystem for Linux" в настройках Windows и установить нужный дистрибутив, например, Ubuntu. Далее можно использовать Bash для запуска скриптов и выполнения команд Linux [2].

Создать скрипт можно в любом текстовом редакторе. В начале скрипта необходимо указать интерпретатор, который будет его выполнять в виде строки: `#!/bin/bash`. Далее можно вписывать команды Bash, которые будут выполняться последовательно (см.рис.1):



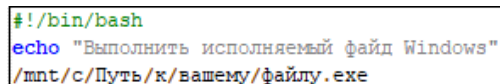
```
#!/bin/bash
echo "Привет, мир!"
mkdir новая_папка
cd новая_папка
touch новый_файл.txt
```

Рис. 1 – Пример простого Bash-скрипта

Для запуска данного Bash-скрипта, сохраненного на Windows, достаточно выполнить команду

`./script.sh` в командной строке WSL, где `script.sh` – имя скрипта.

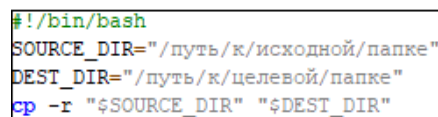
Bash-скрипты могут взаимодействовать с Windows, вызывая исполняемые файлы (см.рис.2), работая с реестром или даже управляя службами.



```
#!/bin/bash
echo "Выполнить исполняемый файл Windows"
/mnt/c/Путь/к/вашему/файлу.exe
```

Рис. 2 – Пример вызова .exe файла из Bash-скрипта

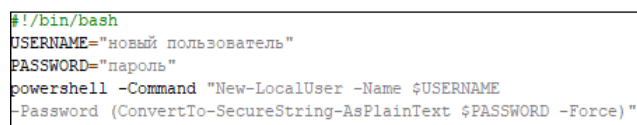
Bash-скрипт, который копирует все файлы из одной папки в другую (см.рис.3):



```
#!/bin/bash
SOURCE_DIR="/путь/к/исходной/папке"
DEST_DIR="/путь/к/целевой/папке"
cp -r "$SOURCE_DIR" "$DEST_DIR"
```

Рис. 3 – Копирование файлов

Использование Bash-скриптов для выполнения команд PowerShell (см.рис.4):



```
#!/bin/bash
USERNAME="новый_пользователь"
PASSWORD="пароль"
powershell -Command "New-LocalUser -Name $USERNAME -Password (ConvertTo-SecureString -AsPlainText $PASSWORD -Force)"
```

Рис. 4 – Создание нового локального пользователя с указанным именем и паролем

II. ВЫВОДЫ

Использование Bash-скриптов на ОС Windows через подсистему Windows для Linux особенно полезно для тех, кто привык к командной строке Linux и хочет использовать ее преимущества в среде Windows.

1. Bash on Windows: What Does It Mean? [Электронный ресурс].–Режим доступа: <https://www.linux.com/news/bash-windows-what-does-it-mean/>.
2. How to Install Linux Bash Shell on Windows [Электронный ресурс].–Режим доступа: <https://itsfoss.com/install-bash-on-windows/>.

Хаджинова Ксения Александровна, студентка группы 320604, кафедра ИТАС, БГУИР, xju2005@gmail.com

Научный руководитель: Тиханович Татьяна Викторовна, зам.декана ФИТиУ, старший преподаватель кафедры ИТАС, БГУИР, tihanovich@bsuir.by.

ИСПОЛЬЗОВАНИЕ МЕТОДОВ ЭКСПЕРТНЫХ ОЦЕНОК ПРИ ПОСТУПЛЕНИИ В ВУЗ

В статье представлен подход, основанный на методах экспертных оценок для ранжирования и окончательного выбора специальности при поступлении в ВУЗ на основании ценностных и целевых установок абитуриента.

ВВЕДЕНИЕ

Для любого человека принятие решений и выбор лучшей альтернативы - весьма важная задача, и последствия этих решений могут определять поведение, действия, а порой и судьбу всей жизни. Для задач многокритериального принятия решений существуют различные методы, выбор которых зависит от особенностей каждой проблемы. Главная проблема при выборе – наличие множества критериев оценки.

I. МЕТОДЫ ЭКСПЕРТНЫХ ОЦЕНОК

Рассмотрим методы экспертных оценок для определения мнений экспертов и формировании на их основе необходимой информации при выборе предпочтений специализации для абитуриента при поступлении в ВУЗ. Данные методы были использованы на основе экспериментальных данных, полученных при поступлении абитуриентов в БГУИР в 2023 году. В роли экспертов являлись абитуриент, родители абитуриента, рейтинг по баллам. В качестве альтернатив – специальности факультета информационных технологий и управления: системы управления информацией (СУИ), информационные системы и технологии (ИСИТ), искусственный интеллект (ИИ), электронные системы и технологии (ЭСИТ).

Метод ранга. Метод основан на балльных оценках альтернатив, указываемые несколькими экспертами. В рамках такой технологии веса альтернатив определяются на основе сведенного в общую матрицу мнения коллектива экспертов, оценивающих важность альтернатив, пользуясь некоторой балльной шкалой оценок (в экспертизе могут использоваться 5-балльная, 10-балльная, 12-балльная, 100-балльная шкалы [1]).

Метод предпочтений. Метод основан на ранжировании альтернатив, выполняемом группой экспертов[1]. Каждый из экспертов выполняет ранжирование альтернатив, т.е. указывает, какая из альтернатив, по его мнению, является лучшей, какая следующей за ней и так далее.

Составляется сводная матрица всех оценок экспертов (см. таблицу 1)

Таблица 1 – Матрица оценок

Эксперты	СУИ	ИСИТ	ИИ	ЭСИТ
Абитуриент	1	3	2	4
Рейтинг по баллам	2	1	3	4
Родитель абитуриента	1	2	4	3

Затем составляется преобразованная матрица оценок, как разница количества альтернатив и текущей оценки (см. таблицу 2)

Таблица 2 – Преобразованная матрица оценок

Эксперты	СУИ	ИСИТ	ИИ	ЭСИТ
Абитуриент	3	1	2	0
Рейтинг по баллам	2	3	1	0
Родитель абитуриента	3	2	0	1

Для расчета наилучшего решения требуется рассчитать веса альтернатив на основе сумм их оценок и суммы всех оценок по формулам: $C_j = \sum_{i=1}^M B_{ij}$ и $C = \sum_{j=1}^N C_j$.

Для расчета весов альтернатив следует воспользоваться формулой: $V_j = C_j/C$. На основе исходных данных наилучшим решением является выбор приоритета среди альтернатив с наибольшим весом: СУИ - ИСИТ - ИИ - ЭСИТ.

II. МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ АБИТУРИЕНТОВ

В результате опроса абитуриентов было выявлено, что многим из них не помешало бы наличие приложения, которое позволяет в доступной форме найти наилучшее решение среди множества альтернатив.

Было разработано приложение для операционной системы Android [2] на основе языка Java, которое позволяет абитуриенту и родителю ввести своё мнение в приложение, а затем получить решение наиболее согласованной между ними.

При запуске приложения абитуриенту и родителю отображается список специальностей с полями для заполнения оценками в диапазоне от 1 до 4, где чем меньше значение, тем приоритетнее специальность (см.рис. 1)

Помощь Абитуриенту

Поставьте специальности в порядке ваших интересов

Абитуриент	Специальности	Родитель
2	Искусственный интеллект (ФИТУ)	4
3	Информационные системы и технологии (ФИТУ)	2
1	Системы управления информацией	1
4	Электронные системы и технологии (ФИТУ)	3

РАСЧЕТ

Рис. 1 – Пример заполнения данных

После заполнения данных требуется произвести расчет по нажатию на кнопку. После чего на экране отобразится оптимальная альтернатива (см.рис.2)

Помощь Абитуриенту

Оптимальная альтернатива:

Системы управления информацией (ФИТУ)

Информационные системы и технологии (ФИТУ)

Искусственный интеллект (ФИТУ)

Электронные системы и технологии (ФИТУ)

Рис. 2 – Расчет оптимального варианта

Хващевский Максим Анатольевич, студент кафедры информационных технологий автоматизированных систем, makshvashevskiy@gmail.com.

Протченко Екатерина Владимировна, аспирант кафедры электронных вычислительных машин, protchenko@bsuir.by.

Научный руководитель: Протченко Екатерина Владимировна, Старший преподаватель кафедры информационных технологий автоматизированных систем БГУИР, protchenko@bsuir.by.

Результатом вычисления программой оптимальной альтернативы стала альтернатива: СУИ - ИСиТ - ИИ - ЭСиТ.

III. Выводы

Согласованность решений в жизни любого человека играет важную роль, особенно выборе будущей профессии у абитуриентов. Методы экспертных оценок открывают огромные возможности для быстрого и правильного согласования действий. Методы для согласованности в действиях и выбора наилучшего решения могут использоваться не только в выборе специальности, но и в других аспектах жизни. Благодаря этим методам абитуриенты смогут согласовать мнение с родителями без особых проблем, что позволит

1. Системный анализ и исследование операций: оптимизация решений на основе методов и моделей математического программирования / С. С. Смородинский, Н. В. Батин – Минск: БГУИР – 2010. – 192с.
2. Дейтел. П. Android для программистов. Создаем приложение / П. Дейтел [и др.]. – М.: Питер, 2020. – 560с.

Секция «Интеллектуальные информационные технологии»

Председатель:	д-р техн. наук, проф., профессор каф. ИИТ Голенков В. В.
Члены жюри:	канд. техн. наук, доцент, зав. каф. ИИТ Шункевич Д. В. канд. физ.-мат. наук, доцент, доцент каф. ИИТ Гулякина Н. А. канд. техн. наук, доцент кафедры ИИТ Захаров В. В.
Секретарь	ст. преподаватель Гракова Н.В.

ГИБРИДНАЯ ОПТИМИЗАЦИЯ КАРПУЛИНГОВЫХ СИСТЕМ ДЛЯ УСТОЙЧИВОЙ ГОРОДСКОЙ МОБИЛЬНОСТИ

Статья рассматривает гибридный подход к многокритериальной оптимизации (МОО) в системе карпулинга. Применение комбинированных алгоритмов позволяет находить сбалансированные решения между временем в пути, загрузкой транспорта и экономической эффективностью, адаптируясь к динамичным условиям городской мобильности.

ВВЕДЕНИЕ

Карпулинг представляет собой форму совместных поездок, при которой пассажиры делят маршрут и транспортное средство. Это решение способствует снижению нагрузки на транспортную систему и уменьшению выбросов. Однако для эффективного функционирования таких систем необходима оптимизация множества параметров. Многокритериальная оптимизация (МОО) используется для одновременного учета нескольких целей, таких как минимизация времени, максимизация заполняемости и обеспечение рентабельности. Гибридные подходы к МОО становятся особенно актуальны в условиях изменчивого городского трафика и индивидуальных предпочтений пользователей.

I. ГИБРИДНЫЙ ПОДХОД К МНОГОКРИТЕРИАЛЬНОЙ ОПТИМИЗАЦИИ В СИСТЕМЕ КАРПУЛИНГА

Гибридный подход к многокритериальной оптимизации объединяет различные алгоритмы — как классические, так и метаэвристические — для достижения комплексного и устойчивого решения [1]. В контексте карпулинга рассматриваются три ключевые цели:

- минимизация времени в пути — $T(x)$;
- максимизация заполняемости транспортных средств — $O(x)$;
- оптимизация тарифной политики — $P(x)$.

Для решения такой задачи используется алгоритмическая структура, включающая:

- оптимизацию маршрутов с использованием алгоритмов Дейкстры, A^* и адаптивных методов в условиях изменяющегося трафика ;

- распределение пассажиров, направленное на эффективную загрузку транспорта, основанное на кластеризации и эвристическом подборе попутчиков ;
- динамическое ценообразование, адаптирующее тарифы под текущий спрос с учётом экономических и поведенческих факторов [2].

В основе гибридной модели лежат методы Парето-оптимизации и генетические алгоритмы, такие как «NSGA-II», которые позволяют находить сбалансированные решения в многомерном пространстве целей. Алгоритм начинает работу с генерации начального множества решений, которые затем эволюционируют при помощи кроссинговера, мутации и отбора. Процесс продолжается до достижения устойчивого фронта Парето [1].

Модель учитывает реалистичные условия городской среды:

- вероятностное распределение спроса;
- динамику дорожного трафика;
- нормативные ограничения;
- поведенческие особенности пользователей — чувствительность к цене, комфорт, время ожидания.

Это обеспечивает не только теоретическую эффективность, но и практическую применимость в реальных условиях [2].

1. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. *A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II* / K. Deb [et al.] // IEEE Transactions on Evolutionary Computation, 2002. – Vol. 6, No. 2. – P. 182–197.
2. Yan, C., Zhu, H., Korolko, N., Woodard, D. *Dynamic Pricing and Matching in Ride-Hailing Platforms* / C. Yan [et al.] // Naval Research Logistics (NRL), 2019. – Vol. 67, No. 8. – P. 705–724.

Али Марван Абдулкарим, Магистрант кафедры экономической информатики БГУИР, maarwan.a.ali@gmail.com.

Научный руководитель: Федосенко Владимир Алексеевич, доцент кафедры экономической информатики БГУИР, кандидат технических наук, доцент, fedosenko@bsuir.by.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПЛАТФОРМЫ OSTIS И ПОПУЛЯРНЫХ ФРЕЙМВОРКОВ ДЛЯ РАЗРАБОТКИ ИИ-АГЕНТОВ

Сравнивается Платформа OSTIS с популярными фреймворками для разработки ИИ-агентов (LangGraph, AutoGen и др.). Анализируются архитектура, представление знаний, управление агентами, интеграция моделей, сильные/слабые стороны.

ВВЕДЕНИЕ

Разработка ИИ-агентов — программ, способных автономно воспринимать, рассуждать и действовать — ключевое направление ИИ. Существуют различные фреймворки для их разработки: LangGraph, AutoGen, CrewAI, LlamaIndex и др. [1]. Платформа OSTIS предлагает альтернативный, знание-ориентированный подход на основе семантических сетей, решающий некоторые их ограничения [2].

I. СРАВНИТЕЛЬНЫЙ АНАЛИЗ

Ключевое отличие Платформы OSTIS заключается в ее фундаменте — Технологии OSTIS, которая использует единую семантическую основу на базе универсального языка — SC-кода для представления как декларативных (факты, онтологии), так и процедурных (алгоритмы, агенты) знаний в виде графовых структур в общей семантической памяти (sc-памяти). Обработка знаний основана на событийно- и агентно-ориентированных моделях.

Другие популярные фреймворки используют иные подходы, часто фокусируясь на интеграции больших языковых моделей (LLM) и оркестрации задач:

1. *LangGraph* — расширение экосистемы LangChain, использует граф-ориентированную архитектуру для определения и выполнения шагов агента как узлов в направленном графе, позволяя создавать *циклические потоки* выполнения и гибко управлять состоянием.
2. *AutoGen* основан на *мультиагентных асинхронных беседах*, где агенты взаимодействуют друг с другом для решения задач, часто требуя минимального вмешательства человека.
3. *CrewAI* применяет ролевой подход, где агенты с определенными ролями и задачами сотрудничают в "команде" для достижения общей цели (*ролевая совместная работа*), поддерживая последовательное и иерархическое выполнение задач.
4. *LlamaIndex* фокусируется на интеграции с данными (Retrieval-Augmented Generation — RAG), предоставляя инструменты для *индексации и извлечения* информации из больших объемов данных для использования агентами.

5. *Semantic Kernel* использует модульный подход на основе "навыков" (skills) и плагинов, позволяя комбинировать ИИ-функции и обычный код, ориентирован на *интеграцию в корпоративные среды*, особенно в экосистеме Microsoft (.NET).

Для систематизации и наглядного представления различий между Платформой OSTIS и рассмотренными фреймворками, их ключевые характеристики сведены в таблицу 1.

Проведенный сравнительный анализ позволяет выделить сильные стороны и ограничения каждого из подходов:

1. Платформа OSTIS эффективна для сложных интегрированных систем (глубокие знания, адаптивность). Другие фреймворки обеспечивают быструю разработку узких задач (автоматизация, RAG) и легкую интеграцию LLM (Python).
2. Платформа OSTIS имеет крутую кривую обучения и меньшее сообщество. Популярные фреймворки зависят от недостатков LLM (непрозрачность, галлюцинации), предлагают поверхностную интеграцию знаний и имеют проблемы масштабирования в комплексных задачах.

ЗАКЛЮЧЕНИЕ

Платформа OSTIS применяет знание-ориентированный подход, в отличие от LLM-центричных/оркестрационных фреймворков. Ее единая семантическая основа и глубокое представление знаний позволяют создавать сложные, семантически совместимые интеллектуальные системы для комплексных задач, требующих интеграции разнородных знаний и моделей.

1. Joshi, S. Review of autonomous systems and collaborative AI agent frameworks / S. Joshi // SSRN Electronic Journal : preprint. — 2025.
2. Zotov, N. Software platform for next-generation intelligent computer systems / N. Zotov // Open Semantic Technologies for Intelligent Systems (OSTIS-2022) : сборник научных статей / БГУИР ; редкол.: В. В. Голенков [и др.]. — Минск, 2022. — Вып. 6. — С. 297–326.

Зотов Никита Владимирович, магистрант, кафедра ИИТ БГУИР, n.zotov@bsuir.by

Научный руководитель: Голенков Владимир Васильевич, д. техн. наук, профессор, golen@bsuir.by

Таблица 1 – Сравнительный анализ Платформы OSTIS и популярных фреймворков ИИ-агентов

Критерий	Платформа OSTIS	Популярные фреймворки
Основная парадигма	Знание-ориентированная. Интеграция разнородных знаний и моделей на базе семантических сетей (SC-код). Цель – создание <i>семантически совместимых</i> интеллектуальных систем.	LLM-центричная / Оркестрационная. Оркестрация LLM, мульти-агентное взаимодействие (через чат, роли, графы), интеграция с данными (RAG), использование внешних инструментов.
Представление знаний	Глубокое, унифицированное. Семантические сети (SC-код) для декларативных и процедурных знаний. Спецификации агентов — часть единой базы знаний.	Различное, часто неформальное или специфичное для задачи. Состояние в графах (LangGraph), неявное в LLM/беседах (AutoGen), векторные базы/индексы (LlamaIndex), объекты состояния, структурированный вывод (JSON/Pydantic). Обычно нет <i>единой формальной семантической основы</i> .
Интеграция моделей решения задач	Глубокая, семантическая. Интеграция различных методов (декларативных, процедурных, логического вывода, машинного обучения) на общей семантической основе – SC-коде.	Поверхностная/Модульная. Фокус на одном подходе (workflow, chat, RAG) или комбинация через API/плагины/вызовы функций без глубокой семантической интеграции.
Управление агентами	Событийно/агентно-ориентированная. Агенты реагируют на события в <i>sc-памяти</i> (изменения в графе знаний) и выполняют действия согласно спецификации (статической/динамической/полудинамической).	Разнообразная. Явные графы потоков работ (LangGraph), асинхронные беседы/сообщения (AutoGen), роли и задачи в "команде" (CrewAI), плагины/навыки (Semantic Kernel), вызов функций/инструментов LLM (OpenAI SDK и др.).
Гибкость и масштабируемость	Высокая. Модульная архитектура, унифицированное представление знаний и нацеленность на параллелизм способствуют масштабированию и адаптации сложных систем.	Различается. LangGraph предлагает масштабируемую инфраструктуру для потоков работ. Другие могут иметь ограничения при интеграции или в сложных сценариях. Существуют общие проблемы масштабирования и управления контекстом для AI агентов на базе LLM.
Память	Встроенная семантическая. <i>sc-память</i> как единая графовая база знаний для всей системы, обеспечивающая долговременную и контекстуальную память.	Явная/Внешняя/Зависимая от LLM. Управление состоянием в графе (LangGraph), краткосрочная/долгосрочная память (CrewAI), интеграция с векторными БД для RAG (LlamaIndex), управление памятью в Semantic Kernel, контекстное окно LLM.
Простота использования / Порог входа	Высокий. Требуется изучения Технологии OSTIS, SC-кода и специфической архитектуры. Более крутая кривая обучения.	Различается. Некоторые фреймворки, особенно с GUI-инструментами (например, AutoGen Studio) или ориентированные на роли (CrewAI), считаются более интуитивными. Другие (LangGraph, Semantic Kernel) могут быть сложнее. Часто используется знакомый язык Python и экосистема вокруг LLM.

МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ЭЛЕКТРОКАРДИОГРАММ

Представлен обзор современных алгоритмов анализа электрокардиограмм с целью выявления патологий сердца. Приведено сравнение алгоритмов определения элементов ЭКГ. Представлен обзор открытых датасетов для разработки алгоритмов анализа ЭКГ.

ВВЕДЕНИЕ

В современной медицине анализ электрокардиограммы (ЭКГ) является одним из ключевых инструментов диагностики сердечно-сосудистых заболеваний. Развитие технологий глубокого обучения открыло новые возможности для автоматизации и повышения качества анализа ЭКГ-сигналов. Последние достижения в области глубокого обучения позволили создавать системы анализа ЭКГ с точностью, сопоставимой с опытными кардиологами [1]. Рассмотрим некоторые из ключевых решений, представленных в научной литературе.

I. ОТКРЫТЫЕ ДАТАСЕТЫ ЭКГ

Одним из ключевых ресурсов для тестирования алгоритмов анализа ЭКГ является ресурс PhysioNet [2]. Ресурс содержит датасеты, включающие записи патологий. Особенно важным является наличие аннотаций во многих датасетах, что делает возможным применение методов анализа, для которых характерно обучение с учителем. Во многих работах по теме анализа ЭКГ присутствует ссылка на открытый датасет MIT-BIH Arrhythmia Database [2], содержащий записи ЭКГ в двух отведениях. К пригодным для анализа можно отнести датасет записей холтеровского мониторинга в 12 отведениях компании ИНКАРТ [2]. Отдельно можно отметить наличие специализированных датасетов, позволяющих анализировать конкретные элементы ЭКГ. Это достигается за счёт наличия соответствующей разметки в датасете. Яркий пример – датасет QT Database [3], в котором размечены P, QRS, T и U. Аналогично датасет European ST-T Database [4] содержит разметку сегмента ST и зубца T. Присутствуют датасеты, содержащие размеченные записи определённых патологий. Так, в датасете кардиологического центра университета Крейтон [5] содержатся записи желудочковой тахикардии, трепетания и фибрилляции желудочков.

II. УЛУЧШЕНИЕ КАЧЕСТВА СИГНАЛА

Методы предобработки и улучшения качества сигналов ЭКГ важны для повышения точности анализа. Один из таких методов представлен в работе Волосатовой и Малышева [6] – это алгоритм удаления дрейфа изолинии, позволяющий устранить низкочастотные артефакты за счёт вычитания из сигнала кубического сплайна, построенного по серединам R-R интервалов. Демонстрация работы алгоритма представлена на рисунке 1.

В исследовании Kwon et al. [7] анализируется частота дискретизации ЭКГ, оптимальная для анализа variability сердечного ритма. Итоговая оценка важна для алгоритмов, чувствительных к размерности входных данных (например, классификация сердечных сокращений с помощью нейросетей, где количество измерений в образце фиксировано).

III. ОБНАРУЖЕНИЯ ЭЛЕМЕНТОВ ЭКГ

Перед анализом обязательно осуществляется обнаружение элементов ЭКГ, таких как зубцы, комплексы и другие. В своей работе Величко и Кампос Диас [8] предложили метод определения QRS-комплексов без предварительной обработки. Алгоритм основан на вейвлет-преобразовании. Результат работы алгоритма представлен на рисунке 2. Wu et al. разработали подход [9], использующий анализ изменчивости окна для усиления сигнала ЭКГ и улучшения детектирования R-пиков. Метод показал высокую устойчивость к шумам.

IV. ДИАГНОСТИКИ ПАТОЛОГИЙ

Заключительным этапом работы алгоритмов анализа ЭКГ является диагностика различных патологий, таких как аритмии и другие заболевания миокарда. Так, в работе [10] рассмотрена модель сверточной нейронной сети для классификации сигналов ЭКГ и выявления аритмий. Данное исследование показывает высокую точность при эффективной обработке больших объёмов данных. В схожей по цели работе Lai et al. [1] предложили алгоритм самонаблюдаемого обучения на масштабируемом наборе данных ЭКГ, что позволило улучшить распознавание патологий. Wu и Guo [11] провели обзор методов глубокого обучения в анализе ЭКГ и диагностике сердечно-сосудистых заболеваний.

V. ВЫВОДЫ

В открытом доступе присутствует множество данных, которые могут быть использованы для разработки и тестирования новых алгоритмов анализа ЭКГ. Также существует множество различных подходов и алгоритмов как обнаружения элементов ЭКГ, так и улучшения качества сигнала и обнаружения патологий. Эффективность алгоритмов подтверждает целесообразность дальнейшего развития интеллектуальных систем анализа ЭКГ. Отдельного внимания заслуживает применение нейросетевых методов в связи с бурным развитием таковых

на момент написания статьи. Разработки, представленные в описанных статьях, будут использованы для дальнейшей работы по перспективному (и менее распространённому на данный момент) направлению интеллектуального анализа кардиограмм новорождённых.

1. Lai, J., Tan, H., Wang, J. et al. Practical intelligent diagnostic algorithm for wearable 12-lead ECG via self-supervised learning on large-scale dataset. Nat Commun 14, 3741 (2023).
2. Goldberger, A., et al. "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation [Online]. 101 (23), pp. e215–e220." (2000).
3. Laguna P, Mark RG, Goldberger AL, Moody GB. A Database for Evaluation of Algorithms for Measurement of QT and Other Waveform Intervals in the ECG. Computers in Cardiology 24:673-676 (1997).
4. Taddei A, Distant G, Emdin M, Pisani P, Moody GB, Zeelenberg C, Marchesi C. The European ST-T database: standard for evaluating systems for the analysis of ST-T changes in ambulatory electrocardiography. Eur Heart J. 1992 Sep;13(9):1164-72.
5. Nolle FM, Badura FK, Catlett JM, Bowser RW, Sketch MH. CREI-GARD, a new concept in computerized arrhythmia monitoring systems. Computers in Cardiology 13:515-518 (1986).
6. Волосатова Т.М., Малышев А.П. Улучшение сигнала электрокардиограммы на основе алгоритма удаления дрейфа его изолинии // Интернет-журнал «НАУКОВЕДЕНИЕ» Том 9, №4 (2017)
7. Kwon O, Jeong J, Kim HB, Kwon IH, Park SY, Kim JE, Choi Y. Electrocardiogram Sampling Frequency Range Acceptable for Heart Rate Variability Analysis. Healthc Inform Res. 2018 Jul;24(3):198-206. doi: 10.4258/hir.2018.24.3.198. Epub 2018 Jul 31. PMID: 30109153; PMCID: PMC6085204.
8. Метод определения QRS-комплексов в ЭКГ-сигналах без предварительной обработки / Величко О.Н., Кампос Диас Х.Г. // Вестник НТУ "ХПИ". Серия: Информатика и моделирование. – Харьков: НТУ "ХПИ". – 2019. – № 28 (1353). – С. 142 – 152.
9. Wu, L.; Xie, X.; Wang, Y. ECG Enhancement and R-Peak Detection Based on Window Variability. Healthcare 2021, 9, 227. <https://doi.org/10.3390/healthcare9020227>
10. ECG Classification System for Arrhythmia Detection Using Convolutional Neural Network [электронный ресурс] Режим доступа: <https://arxiv.org/html/2303.03660v2>. – Дата доступа: 29.03.2025.
11. Wu, Z., Guo, C. Deep learning and electrocardiography: systematic review of current techniques in cardiovascular disease diagnosis and management. BioMed Eng OnLine 24, 23 (2025).

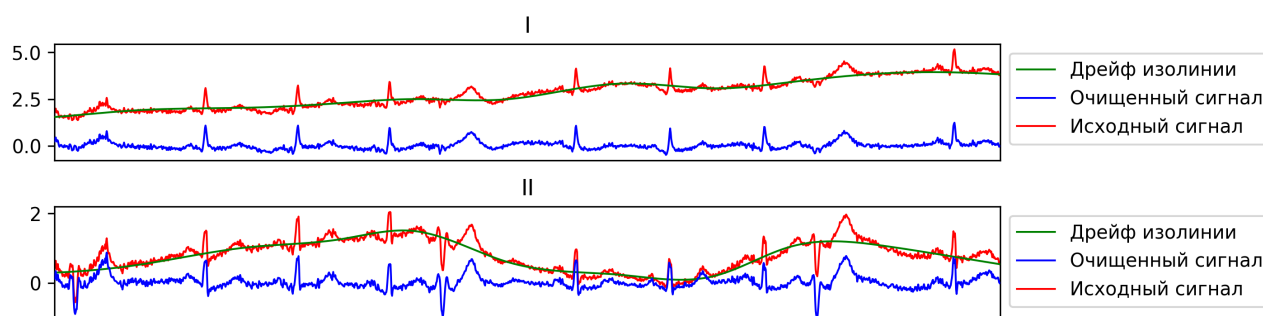


Рис. 1 – Удаление дрейфа изолинии сигнала ЭКГ по отведениям

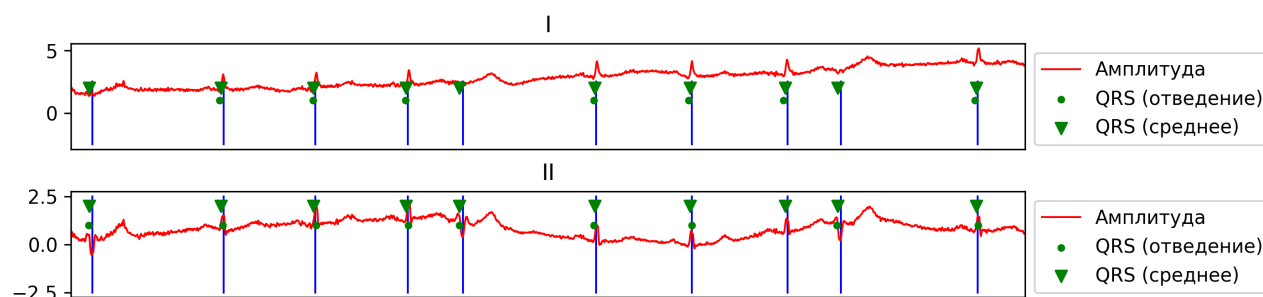


Рис. 2 – Обнаружение QRS комплексов на ЭКГ по отведениям, сравнение с разметкой датасета

Робилко Тимур Маркович, студент кафедры интеллектуальных информационных технологий БГУ-ИР, timur.robilko@gmail.com.

Научный руководитель: Ковалёв Михаил Владимирович, магистр технических наук, старший преподаватель кафедры интеллектуальных информационных технологий БГУИР, kovalev@bsuir.by.

ПРИМЕНЕНИЕ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ВОССТАНОВЛЕНИЯ ЗАШУМЛЕННОЙ ИНФОРМАЦИИ

В данной статье рассматривается возможность восстановления зашумленной информации с помощью рекуррентных нейронных сетей. Определены источники шума, его классификация, рассмотрены доступные методы восстановления информации. Представлены примеры задач восстановления зашумленной информации, решения которых возможно с помощью рекуррентных нейронных сетей. Выделены преимущества и недостатки их использования, описаны примеры успешной реализации. Выдвинуто предположение относительно дальнейших шагов в исследовании данного направления.

ВВЕДЕНИЕ

В нейронной сети каждый нейрон принимает входные данные, обрабатывает их с помощью весов и активационной функции, а затем передает результат на следующий уровень. Такой подход позволяет сети обучаться на основе примеров, адаптируя свои параметры для улучшения точности предсказаний [1]. Это делает нейронные сети мощным инструментом, позволяя моделировать им сложные зависимости и паттерны в данных. Они служат основой для разных приложений на основе технологий искусственного интеллекта, включая средства распознавания изображений, обработку естественного языка и генерацию контента.

Существуют различные типы нейронных сетей. Одним из них являются рекуррентные сети, где выход сети зависит от предыдущих вычислений [2].

Шум в данных – это случайные флуктуации значений признаков, которые не несут полезной информации. Причинами возникновения шума могут быть ошибки измерений, помехи в процессе передачи данных и переизбыток информации. Зашумленная информация может затруднять анализ, искажать результаты и приводить к неправильным решениям, принятым на основе неверных данных.

Среди основных подходов к восстановлению зашумленной информации можно выделить применение нейронных сетей, что широко используется для восстановления зашумленных сигналов и изображений.

Архитектура рекуррентных нейронных сетей основана на последовательном распространении информации, что делает их особенно подходящими для задач, связанных с временными рядами и последовательностями, такими как обработка естественного языка [1]. Это позволяет рекуррентным нейронным сетям учитывать контекст и предшествующие состояния, что критично для восстановления информации из зашумленных данных. Такая архитектурная особенность делает рекуррентные сети подходящим инструментом для восстановления зашумленной информации.

I. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ

Концепция рекуррентных нейронных сетей была впервые предложена Дэвидом Румельхартом, в

журнале Nature в 1986 году, для описания новой процедуры обучения [3]. Цель ее разработчиков заключалась в имитации человеческой памяти.

Рекуррентные нейронные сети ориентированы на задачи, связанные с временными рядами и последовательностями. Обеспечение эффективной обработки последовательных данных является основной целью рекуррентных нейронных сетей.

Архитектура рекуррентных сетей, как и архитектура других нейронных сетей, включает в себя нейроны, слои и связи. Связи между нейронами формируют направленную последовательность, что позволяет им обрабатывать последовательные данные.

Нейроны являются основными строительными блоками любой нейронной сети, они обрабатывают входные данные и передают результаты. В рекуррентных сетях нейроны могут иметь связи не только с нейронами в предыдущем слое, но и с нейронами в том же слое [4]. Такая архитектура позволяет учитывать информацию из предыдущих временных шагов.

Для определения выхода на основе входных данных, нейроны могут использовать различные функции активации. Функции активации вводят нелинейность, ограничивают амплитуду выходного сигнала нейрона, и способны влиять на скорость обучения. В рекуррентных нейронных сетях используются различные активационные функции, включая сигмоиду, гиперболический тангенс и линейную функцию. Каждая из этих функций имеет свои особенности и может быть выбрана в зависимости от конкретной задачи и архитектуры сети.

Рекуррентные нейронные сети, как и другие вариации нейронных сетей, состоят из входного, скрытых и выходного слоев. Они обрабатывают входные данные и передают результаты на следующий уровень. Организация скрытых слоев позволяет сети запоминать информацию о предыдущих входах, и изменять веса не только в зависимости от входных данных, но и от предыдущих выходов. Рекуррентная нейронная сеть разворачивается в полную сеть. Если последовательность представляет собой предложение из некоторого количества слов, то развертка будет состоять из количества слоев, равного количеству слов в предложении.

Связи в нейронных сетях представляют собой математические отношения между нейронами, ко-

которые позволяют передавать информацию от одного нейрона к другому [1]. Сигнал в рекуррентных нейронных сетях движется не только в прямом направлении, но и в обратном. Как следствие, результат выхода способен возвращаться на вход. Выход нейрона определяется весовыми характеристиками и входными сигналами, что также дополняется предыдущими выходами. За счет изменения веса не только в зависимости от входных данных, но и от предыдущих выходов, нейросетям присуща функция кратковременной памяти, что позволяет сигналам восстанавливаться и дополняться во время их обработки.

Такая архитектура позволяет рекуррентным сетям эффективно обрабатывать данные, где важен порядок. Например, в задачах обработки естественного языка, данные могут учитывать контекст слов в предложении, что способствует улучшению качества предсказаний.

В отличие от сверточных нейронных сетей, которые эффективно извлекают пространственные зависимости в изображениях и видео, рекуррентные нейронные сети способны учитывать порядок и контекст входных данных [5]. Это адаптирует их для задач обработки естественного языка и анализа временных рядов. Полносвязные нейронные сети не могут учитывать последовательность данных, что ограничивает их применение в ситуациях, где порядок имеет значение.

Обучение нейронных сетей – это процесс, в ходе которого модель искусственного интеллекта учится выполнять определенные задачи на основе предоставленных ей данных и меток, из которых можно сделать вывод о правильности обучения. Сеть учится на примерах и постепенно улучшает свои предсказания, минимизируя ошибки. Подход к обучению зависит от разных факторов, что включают цель обучения, тип решаемой задачи, характеристику и размер данных, выбранную архитектуру.

Обучение рекуррентных нейронных сетей является сложной задачей, так как они подвержены проблеме исчезающих градиентов, что приводит к медленному или нестабильному обучению, и трудностям в изучении долгосрочных зависимостей. Основополагающим методом для обучения рекуррентных нейронных сетей является обратное распространение ошибки, который позволяет оптимизировать веса и улучшать предсказательную способность модели. Этот метод основан на градиентном спуске, минимизирующем функцию потерь, которая измеряет, насколько хорошо модель предсказывает выходные данные.

Для решения проблемы затухания градиентов, на базе стандартной архитектуры, было разработано несколько расширенных вариантов, основанных на принципе сохранения памяти.

Долгая краткосрочная память (LSTM) является вариацией рекуррентной нейронной сети. Ее назначением является решение проблемы исчезающего градиента. В данной архитектуре увеличен объем

памяти для размещения данных за более длительный период, что позволяет эффективно запоминать долгосрочные зависимости [6].

Управляемые рекуррентные блоки (GRU) – это вариация, являющаяся упрощенной версией LSTM, также поддерживающая выборочное сохранение памяти. Данная модель объединяет некоторые функции и имеет меньшую вычислительную сложность. Обязательным элементом ее скрытого слоя являются плюзы, позволяющие хранить информацию в памяти и удалять ее [7]. Выбор между описанными архитектурами зависит от конкретной задачи и требований к модели. LSTM и GRU эффективнее работают с памятью и позволяют лучше справляться с долгосрочными зависимостями в данных. Стандартная модель больше подходит для простых задач, где не требуется запоминание долгосрочных зависимостей.

Рекуррентные нейронные сети представляют собой особый тип нейронных сетей, которые предназначены для обработки последовательных данных. В отличие от традиционных нейронных сетей, где информация передается только в одном направлении, рекуррентные сети имеют циклические связи, что позволяет им сохранять информацию о предыдущих входах и использовать ее для предсказания следующих значений в последовательности.

II. ПОНЯТИЕ ЗАШУМЛЕННОЙ ИНФОРМАЦИИ

Развитие информационных технологий ведет к увеличению объемов данных и источников информации. Огромное количество информации образуется в таких источниках, как социальные сети, интернет вещей и медицинская диагностика. Накопление информации требует решения задач фильтрации шума и восстановления полезного сигнала.

Шум в данных – это случайные флуктуации значений признаков, которые не несут полезной информации и могут иметь различные источники. Причинами возникновения шума являются ошибки измерений, помехи в процессе передачи данных и переизбыток информации, что ведет к трудностям в анализе и интерпретации данных. Классификация шума способствует пониманию того, как он влияет на восприятие информации. В контексте обработки сигналов, шум представляет собой составляющую сигнала, имеющую случайный характер.

Основные источники шума могут включать в себя внешние и внутренние факторы. Внешние источники могут быть связаны с условиями сбора данных, такими как плохое освещение для изображений или фоновый шум в аудиозаписях. Внутренние источники шума могут возникать из-за ошибок в алгоритмах обработки данных или неправильной настройки параметров модели.

Важной характеристикой является соотношение сигнала и шума, которое определяет силу полезного сигнала относительно фонового шума. Это соотношение критично для оценки качества передачи и обработки данных.

Шум в данных может быть случайным и систематическим. Случайный шумом называется тот, который имеет случайный характер и не несет полезной информации. Систематический шум, в отличие от случайного, представляет собой предсказуемые и повторяющиеся ошибки, и может возникать из-за неправильной калибровки оборудования или других систематических факторов, что способны привести к систематическим ошибкам в выводах. Оба типа шума могут существенно влиять на качество данных и результаты их анализа, поэтому важно применять соответствующие методы для их идентификации и устранения.

Зашумленной информацией называется информация, содержащая элементы, искажающие ее смысл или усложняющие понимание. Данный термин часто используется для обозначения информационного шума, который описывает наличие ненужных данных, мешающих восприятию смысла.

В более широком смысле, зашумленность может относиться к любым факторам, которые ухудшают качество передачи или восприятия информации. Это может включать технические помехи, ошибки в данных или даже культурные и языковые барьеры [6]. Одним из основных типов является информационный шум, который представляет собой элементы, усложняющие понимание текста и искажающие его смысл. Этот шум может возникать из-за избыточной информации, неуместных данных или нечетких формулировок, что затрудняет восприятие и анализ информации.

К распространенным типам искаженной информации относятся: искажения или помехи в аудиозаписях, потеря четкости видео, шум на спутниковых изображениях и результатах МРТ, искажение данных из-за ошибок измерений и случайных колебаний, а также аномалии, возникающие в процессе передачи информации по каналам связи, вызванные электромагнитными помехами и внутренними ошибками в кодировании. Каждый из этих типов зашумленной информации требует специфических методов для восстановления и очистки.

Зашумленная информация может затруднять анализ, искажать результаты и приводить к неправильным решениям, принятым на основе неверных данных. Это обосновывает актуальность процесса фильтрации и очистки информации для достижения более точных результатов.

III. ВОССТАНОВЛЕНИЕ ЗАШУМЛЕННОЙ ИНФОРМАЦИИ

Восстановление зашумленной информации – это процесс, направленный на улучшение качества данных, которые были искажены или затенены шумом.

Существует несколько методов и алгоритмов, которые могут быть использованы для этой цели, среди них можно выделить:

- алгоритмы фильтрации;
- статистические методы;

- алгоритмы машинного обучения.

Одним из основных подходов является использование фильтров, предназначенных для обработки данных с целью удаления шума или выделения полезной информации. Они могут быть как простыми, так и сложными.

Алгоритмы фильтрации могут использоваться в различных направлениях, включая обработку сигналов, компьютерное зрение и анализ временных рядов. Примерами являются фильтры Калмана и медианные фильтры, способные уменьшить шум и сохранить важные детали информации.

Статистические методы основываются на статистических моделях и теории вероятностей, и используются для анализа данных, выявления закономерностей и принятия решений на основе данных. Они могут включать в себя методы, такие как регрессионный анализ, байесовские методы, методы максимального правдоподобия. Данные методы применены для оценки вероятности различных состояний данных и выбора наиболее вероятного.

Нейронные сети широко используются для восстановления зашумленных сигналов и изображений. Распознавание паттернов и восстановление оригинальных данных возможно после обучения сетей на больших наборах данных. Методы, основанные на моделях, используют предположения о структуре данных для восстановления. Для примера, сверточные нейронные сети часто используются для обработки изображений, позволяя эффективно удалять шум и улучшать качество [5].

Указанные методы и алгоритмы могут быть использованы в различных областях, включая обработку сигналов, анализ данных и восстановление зашумленной информации. Важным шагом в процессе восстановления информации является выбор подходящего метода, который зависит от типа данных и характера шума. Корректный выбор позволит достичь наилучших результатов.

IV. ПРИМЕНЕНИЕ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ

Машинное обучение играет ключевую роль в восстановлении зашумленной информации. Методы машинного обучения являются высокоэффективными инструментами для обработки и анализа данных. Эти методы активно развиваются и применяются в различных областях, включая обработку данных и телекоммуникации. Восстановление зашумленных данных с использованием рекуррентных нейронных сетей предполагает несколько ключевых шагов:

- сбор и подготовку данных;
- создание модели;
- обучение модели;
- восстановление данных;
- оценку производительности.

Данные играют ключевую роль в обучении. Они должны быть предварительно подготовлены для обучения, и содержать как чистые, так и зашумленные примеры. Это могут быть временные ряды, аудиосигналы или другие последовательные

данные. Их обработка подразумевает нормализацию и разделение данных на обучающую и тестовую выборки.

Рекуррентные нейронные сети могут иметь различную архитектуру. Выбор архитектуры зависит от сложности задачи и характера данных. Важным шагом является определение как количества слоев, так и количества нейронов в каждом слое. Более сложные архитектуры способны лучше справляться с шумом, но требуют больше вычислительных ресурсов.

Целью обучения является минимизация разницы между восстановленными и чистыми данными. Подобный эффект достижим с помощью функции потерь. В процессе обучения модель учится выявлять паттерны в данных и игнорировать шум, используя информацию из предыдущих временных шагов.

Результатом завершения процесса обучения будет являться готовность модели к использованию для восстановления данных, что предполагает поступление на вход сети зашумленных данных, и последующую генерацию предсказаний, представляющих собой восстановленные данные. Для достижения наилучших результатов может потребоваться дополнительная обработка выходных данных, то есть фильтрация или сглаживание.

По завершению процесса восстановления данных необходимо оценить качество результатов с помощью различных метрик, таких как соотношение уровней сигнала и шума, или структурное сходство. Сравнение восстановленных данных с оригинальными позволит определить эффективность обученной модели.

На основе результатов оценки можно внести изменения в архитектуру модели, параметры обучения или методы обработки данных для улучшения качества восстановления. Возможно комбинирование рекуррентных нейронных сетей с другими типами нейронных сетей для повышения точности.

Восстановление зашумленных данных с помощью рекуррентных нейронных сетей – это итеративный процесс, который требует тщательной настройки и оценки на каждом этапе.

ЗАКЛЮЧЕНИЕ

Рекуррентные нейронные сети представляют собой вариацию архитектуры нейронных сетей. Циклические связи отличают их от других нейронных сетей, позволяя учитывать предшествующую информацию, что делает рекуррентные нейронные сети актуальными для задач, связанных с временными рядами и последовательностями. Рекуррентные сети имеют множество преимуществ для восстановления зашумленной информации, включая обработку последовательностей переменной длины, способность к запоминанию контекста и адаптивность.

Основным классом задач, которые решаются рекуррентными нейронными сетями, является обработка текстов [8]. Обучаясь на больших объемах

данных, модели способны создавать логичные тексты, восстанавливать пропущенные слова или фразы в тексте с учетом контекста и структуры предложений. Они эффективно обрабатывают последовательности слов, учитывая контекст и порядок, что позволяет создавать более точные модели для понимания и генерации языка.

Однако, рекуррентные нейронные сети также сталкиваются с определенными ограничениями, среди которых были выделены: проблемы с долгосрочной зависимостью, сложность обучения, затухающий градиент. Данные ограничения способны замедлить процесс восстановления данных.

Рекуррентные нейронные сети отличаются от других типов нейронных сетей по нескольким ключевым аспектам:

- ориентированы на обработку последовательных данных;
- позволяют учитывать временные зависимости;
- используют обратные связи для адаптации к изменениям во входных данных;
- имеют внутреннюю память.

Существует множество успешных проектов и исследований, где задействованы рекуррентные нейронные сети, включая проекты компаний Google и Apple. Рассматриваемая технология способствует повышению качества перевода, позволяет учитывать контекст и структуру данных. Также она применяется для прогнозирования цен на акции, анализа климатических данных или диагностики здоровья пациентов.

В дальнейшем предполагается практическая реализация шагов, описанных в данной статье, то есть обучение модели и проведения эксперимента по восстановлению данных. Ожидается, что результаты эксперимента позволят оценить эффективность различных архитектур рекуррентных нейронных сетей и выявить оптимальные подходы для восстановления зашумленной информации.

Таким образом, благодаря своей способности обрабатывать последовательные данные и учитывать контекст, рекуррентные нейронные сети представляют собой перспективный подход для восстановления зашумленной информации. Необходимость обработки растущих объемов данных и популяризация технологий, включая IoT и большие данные, позволяет сделать вывод об актуальности рассмотренного направления как сейчас, так и в будущем.

Список литературы

1. Haykin, S. Neural Networks: A Comprehensive Foundation / S. Haykin // Prentice Hall, 1998. – P. 842.
2. Krauss, P. Artificial Intelligence and Brain Research / P. Krauss // Springer Berlin, 2024. – P. 260.
3. Rumelhart, D. Learning representations by back-propagating errors / D. Rumelhart, G. Hinton, R. Williams // Nature. – 1986. – Vol. 323. – P. 533–536.
4. Tyagi, A. Recurrent Neural Networks: Concepts and Applications / A. Tyagi, A. Abraham // CRC, 2022. – P. 396.

5. Patil, V. Restoration of Images using only noisy data A Deep Learning Approach using U-NET / V. Patil, P. Katte // IJRAR. – 2019. – Vol. 6. – P. 988–993.
6. Maher, S. Recurrent Neural Networks for Chatbot Excellence: Examining the Power of LSTM Architecture / S. Maher, R. Gaikwad, S. Nimbhore // ICT for Intelligent Systems. – 2024. – P. 513–522.
7. Das, S. Recurrent Neural Networks (RNNs): Architectures, Training Tricks, and Introduction to Influential Research / S. Das, A. Tariq, T. Santos // Machine Learning for Brain Disorders. Neuromethods. – 2023. – Vol. 197. – P. 117–138.
8. Lai, S. Recurrent Convolutional Neural Networks for Text Classification / S. Lai, K. Lui, J. Zhao // AAAI Conference on Artificial Intelligence. – 2015. – P. 2267–2273.

Славинский Григорий Александрович, магистрант кафедры информатики Белорусского государственного университета информатики и радиоэлектроники, gr.slavin.sci@mail.ru.

Научный руководитель: Боброва Наталья Леонидовна, доцент кафедры информатики Белорусского государственного университета информатики и радиоэлектроники, bobrova@bsuir.by.

МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО ПОИСКА, АДАПТИВНОЙ СОРТИРОВКИ И РЕКОМЕНДАЦИЙ НА МУЗЫКАЛЬНЫХ СТРИМИНГОВЫХ СЕРВИСАХ

В работе рассматриваются основные сведения и определения алгоритмов интеллектуального поиска, адаптивной сортировки и рекомендаций в музыкальных сервисах. Приводится краткое описание работы этих алгоритмов и их характеристик, а также их применение в современных стриминговых платформах. Проведен сравнительный анализ нескольких методов, используемых для улучшения пользовательского опыта.

ВВЕДЕНИЕ

Музыкальные платформы используют алгоритмы для сортировки плейлистов, персонализированных рекомендаций и интеллектуального поиска [1]. Эти алгоритмы анализируют поведение пользователей, частоту прослушивания и аудиофичи треков для обеспечения релевантных результатов [2].

I. АЛГОРИТМ АДАПТИВНОЙ СОРТИРОВКИ ПЛЕЙЛИСТОВ (DYNAMIC PLAYLIST RANKING, DPR)

DPR — это алгоритм, который динамически изменяет порядок треков в плейлисте на основе частоты их прослушивания и времени последнего воспроизведения [1].

Формула ранжирования трека:

$$S_i = \alpha F_i + \beta T_i \quad (1)$$

где: S_i — итоговый балл трека, F_i — частота воспроизведения, T_i — время с последнего прослушивания (обратная зависимость), α , β — коэффициенты, регулирующие влияние факторов [3].

Основные характеристики:

- Сортировка треков по популярности и давности прослушивания
- Автоматическое перемещение наиболее актуальных треков вверх списка
- Постепенное понижение редко воспроизводимых песен

Применение: используется в персонализированных плейлистах, таких как "Любимые треки" или "Часто слушаемое".

II. АЛГОРИТМ ИНТЕЛЛЕКТУАЛЬНОГО ПОИСКА МУЗЫКИ (SMART MUSIC SEARCH, SMS)

SMS — это метод, который улучшает поиск по названиям треков, учитывая частоту их прослушивания и степень схожести названия с запросом пользователя.

Формула релевантности:

$$R = \gamma C + \delta F + \epsilon L \quad (2)$$

где: R — итоговый рейтинг результата, C — степень соответствия запроса (коэффициент схожести

строк), F — частота прослушивания, L — время последнего воспроизведения, γ , δ , ϵ — регулирующие коэффициенты.

Основные характеристики:

- Использование нечеткого поиска (fuzzy search) для исправления опечаток
- Приоритет популярных и недавно прослушанных треков в выдаче
- Гибкая сортировка результатов на основе пользовательских предпочтений

Применение: используется в поисковых системах музыкальных сервисов для быстрого и точного подбора треков.

III. АЛГОРИТМ ПРЕДСКАЗАНИЯ СЛЕДУЮЩЕЙ ПЕСНИ (NEXT SONG PREDICTION, NSP)

NSP — это алгоритм, основанный на анализе последовательностей прослушивания. Он предсказывает следующую песню, основываясь на истории пользователя.

Формула вероятности выбора следующего трека:

$$P(T_{n+1}|T_n) = \frac{N(T_n, T_{n+1})}{N(T_n)} \quad (3)$$

где: $P(T_{n+1}|T_n)$ — вероятность того, что после T_n будет T_{n+1} , $N(T_n, T_{n+1})$ — количество случаев, когда T_n и T_{n+1} воспроизводились подряд, $N(T_n)$ — общее количество прослушиваний трека T_n .

Основные характеристики:

- Использование моделей Маркова или нейросетей (LSTM) для предсказания
- Подбор треков, которые логически следуют за текущими в плейлисте
- Учет жанровых и темповых предпочтений пользователя

Применение: формирование автоматических плейлистов, "Радио" на основе одной песни.

IV. СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ

При сравнении алгоритмов DPR, SMS и NSP важно учитывать их эффективность, точность, вычислительные затраты и влияние на пользовательский опыт [4,5].

DPR (Dynamic Playlist Ranking)

Преимущества:

- Обеспечивает актуальность плейлистов [1], динамически адаптируясь к предпочтениям пользователя
- Высокая производительность [3] за счет использования взвешенного ранжирования
- Простая реализация [6] и низкие вычислительные затраты

Недостатки:

- Не учитывает внешние факторы (например, настроение пользователя)
- Может не предлагать разнообразие, если пользователь слушает одни и те же треки

SMS (Smart Music Search)

Преимущества:

- Учитывает нечеткое соответствие, исправляя опечатки и расширяя поиск
- Балансирует между релевантностью запроса и популярностью треков

Недостатки:

- Средняя производительность, так как алгоритм требует сложных вычислений
- Может возвращать нерелевантные результаты при неточных запросах

NSP (Next Song Prediction)

Преимущества:

- Позволяет предсказывать последовательность воспроизведения, повышая персонализацию
- Использует машинное обучение и Марковские модели для адаптации к пользователю

Недостатки:

- Низкая производительность из-за сложных вычислений (особенно в нейросетевых моделях)
- Требуется большое количество данных для обучения

Таблица 1 – Сравнение алгоритмов

Критерий	DPR	SMS	NSP	
Скорость	Выс.	Ср.	Низ.	
Точность	Ср.	Выс.	Выс.	

ЗАКЛЮЧЕНИЕ

Алгоритмы интеллектуальной обработки данных играют ключевую роль в развитии музыкальных платформ. DPR обеспечивает удобную сортировку плейлистов, SMS улучшает поиск, а NSP позволяет формировать персонализированные плейлисты. Введение формализованных моделей и алгоритмов повышает точность рекомендаций и удобство использования платформы. Выбор метода зависит от конкретных задач платформы и требований пользователей.

Список литературы

1. Шигео Акаи, Тошио Сирано. Алгоритмы поиска и сортировки данных в системах рекомендаций. М.: Издательство МГТУ им. Баумана, 2015.
2. Гевин Смит. Методы интеллектуальной обработки данных для анализа музыки. Нью-Йорк: Скотт Карнелл, 2020.
3. Джонс М. А., Ли К. В. Методы нечеткого поиска и их применение в музыкальных базах данных. Лондон: Springer, 2016.
4. Ким Х. С., Парк С. Ю. Алгоритмы ранжирования и поиска в больших данных: применение в музыкальных платформах. Берлин: Springer, 2018.
5. Чжан Ю., Ван Л. Модели Маркова и их применение в рекомендательных системах. Пекин: Tsinghua University Press, 2017.
6. Ли Х. Дж., Чен Ю. Т. Глубокое обучение для анализа последовательностей: применение в музыкальных рекомендациях. Сан-Франциско: O'Reilly Media, 2020.

Шерстобитов Никита Александрович, студент 1 курса Факультета компьютерного проектирования Белорусского государственного университета информатики и радиоэлектроники, Nekitosik207@yandex.by.

Зинькович Артемий Владимировна, студент 1 курса Факультета компьютерного проектирования Белорусского государственного университета информатики и радиоэлектроники, zavuloncb1@gmail.com.

Научный руководитель: Рязанцев Дмитрий Дмитриевич, ассистент кафедры вычислительных методов и программирования, d.riazantsev@bsuir.by.

Секция «Вычислительные методы и программирование»

Председатель:	канд. тех. наук, доцент Кукин Д. П.
Члены жюри:	д-р физ.-мат. наук, профессор Колосов С. В. ст. преподаватель Гуревич О. В. ст. преподаватель Шатилова О. О.
Секретарь	ассистент Радченко А. Д.

ВИЗУАЛИЗАЦИЯ ТРЁХМЕРНЫХ СЦЕН МЕТОДОМ ТРАССИРОВКИ ПУТИ С ИСПОЛЬЗОВАНИЕМ ГРАФИЧЕСКОГО API VULKAN

Рассматриваются принципы и особенности реализации алгоритма трассировки пути (path tracing) для фотореалистичной визуализации трёхмерных сцен с использованием низкоуровневого графического API Vulkan. Анализируются преимущества Vulkan для задач рендеринга с интенсивными вычислениями, в частности, возможности расширений для трассировки лучей (Vulkan Ray Tracing).

ВВЕДЕНИЕ

Создание фотореалистичных изображений является одной из фундаментальных задач компьютерной графики. Трассировка пути, основанная на алгоритме Монте-Карло для решения уравнения рендеринга, позволяет достичь высокой степени реализма за счёт точного моделирования глобального освещения. Однако этот метод традиционно связан с высокими вычислительными затратами, что ограничивало его применение в приложениях реального времени.

С появлением современных графических процессоров (GPU) и низкоуровневых API, таких как Vulkan, открылись новые возможности для ускорения сложных алгоритмов рендеринга. Vulkan предоставляет разработчикам прямой контроль над оборудованием, минимизируя накладные расходы драйвера и позволяя эффективно использовать параллельные вычислительные мощности GPU. Особый интерес представляют стандартизированные расширения Vulkan Ray Tracing, которые обеспечивают аппаратную поддержку для ускорения операций пересечения лучей с геометрией [1].

Целью данной работы является анализ подходов к реализации рендеринга методом трассировки пути с использованием возможностей графического API Vulkan, в частности его расширений для трассировки лучей, и обсуждение ключевых аспектов построения соответствующего конвейера визуализации.

I. ОСНОВЫ РЕАЛИЗАЦИИ ТРАССИРОВКИ ЛУЧЕЙ В VULKAN: СТРУКТУРЫ И КОНВЕЙЕР

Реализация трассировки пути средствами Vulkan, особенно с использованием специализированных расширений, включает несколько ключевых этапов. Создание Структур Ускорения (Acceleration Structures - AS). Эффективная трассировка лучей невозможна без пространственных структур данных, ускоряющих поиск пересечений лучей с объектами сцены. Расширение VK_KHR_acceleration_structure определяет два типа AS:

- Структура ускорения нижнего уровня (Bottom-Level Acceleration Structure - BLAS): Строится для каждой уникальной геометрии (меша) в сцене;

- Структура ускорения верхнего уровня (Top-Level Acceleration Structure - TLAS): Строится над

экземплярами (instances) BLAS, представляя всю сцену и учитывая их трансформации.

Создание и обновление AS – вычислительно интенсивные задачи, которые выполняются на GPU с использованием команд Vulkan. Качество и скорость построения AS напрямую влияют на общую производительность рендеринга.

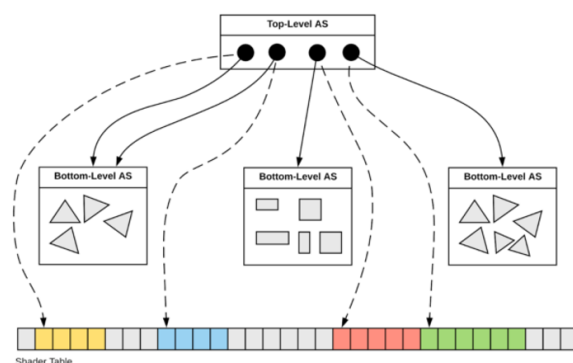


Рис. 1 – Обзор двухуровневой структуры ускорения

Построение Конвейера Трассировки Лучей. Расширение VK_KHR_raytracing_pipeline вводит новый тип графического конвейера, отличный от традиционного растеризационного. Этот конвейер включает специализированные программируемые шейдерные стадии:

- Ray Generation (RGen): Шейдер генерации лучей. Запускается для каждого пикселя изображения, генерирует первичные лучи (например, из камеры). Он иницирует процесс трассировки вызовом `vkCmdTraceRaysKHR`;

- Intersection (Isect): Шейдер пересечения (опциональный). Используется для проверки пересечения луча с пользовательскими примитивами (не только треугольниками);

- Any Hit (AHit): Шейдер любого пересечения (опциональный). Вызывается при каждом обнаружении пересечения луча с примитивом. Может использоваться, например, для обработки полупрозрачных объектов (alpha testing), решая, нужно ли продолжать поиск более близких пересечений;

- Closest Hit (CHit): Шейдер ближайшего пересечения. Вызывается для самого близкого к источнику луча пересечения. Здесь реализуется основная логика path tracing: определение свойств материала поверхности, расчет отражений/преломлений

(на основе BRDF/BSDF), генерация вторичных лучей, сэмплирование источников света и накопление цвета вдоль пути луча;

– Miss (Miss): Шейдер промаха. Вызывается, если луч не пересек ни один объект в сцене. Обычно используется для выборки цвета из карты окружения (environment map) или задания фонового цвета.

Таблица Привязки Шейдеров (Shader Binding Table - SBT). SBT – это область памяти на GPU, которая связывает геометрические примитивы в структурах ускорения с соответствующими группами шейдеров (Hit Group – комбинация Isect, AHit, CHit) и шейдерами Miss. При вызове vkCmdTraceRaysKHR GPU использует индексы из SBT для определения, какие шейдеры выполнять при пересечении луча с конкретным объектом или при промахе. Это обеспечивает гибкость в назначении различных шейдерных программ разным объектам сцены.

II. РЕАЛИЗАЦИЯ АЛГОРИТМА И ПРАКТИЧЕСКИЕ ВЫЗОВЫ

Реализация Алгоритма Трассировки Пути. Основная логика самого алгоритма path tracing реализуется преимущественно в шейдерах Ray Generation и Closest Hit. RGen иницирует трассировку для каждого пикселя. CHit рекурсивно (или итеративно) продолжает путь луча после каждого отскока от поверхности, пока не будет достигнута максимальная глубина трассировки, луч не уйдет в фон (обрабатывается Miss шейдером) или его вклад не станет пренебрежимо малым (например, с использованием техники "русской рулетки"). На каждом шаге происходит накопление световой энергии вдоль пути.

Несмотря на аппаратное ускорение, реализация эффективной трассировки пути в реальном времени остается сложной задачей. Основные вызовы включают:

– Шум: Стохастическая природа метода Монте-Карло приводит к появлению шума на изображении, особенно при малом количестве сэмплов (лучей) на пиксель. Требуются методы шумоподавления (denoising), которые могут быть интегрированы в конвейер рендеринга [1];

– Производительность: Сложность сцены, количество источников света, свойства материалов и глубина трассировки сильно влияют на произво-

дительность. Необходимы оптимизации, такие как важностное сэмплирование (importance sampling) источников света и BRDF, адаптивная глубина трассировки, оптимизация структур ускорения;

– Управление ресурсами: Vulkan требует явного управления памятью, ресурсами (буферы, изображения, AS, SBT) и синхронизацией операций на GPU, что усложняет разработку по сравнению с API более высокого уровня.

III. ВЫВОДЫ

Графический API Vulkan, особенно с его расширениями для трассировки лучей, предоставляет мощную и гибкую основу для реализации алгоритмов глобального освещения, таких как трассировка пути. Низкоуровневый доступ к GPU и аппаратное ускорение ключевых операций позволяют достигать высокой производительности, открывая перспективы для использования фотореалистичного рендеринга в интерактивных приложениях и играх.

Хотя остаются вызовы, связанные с управлением шумом, оптимизацией производительности и сложностью разработки на Vulkan, дальнейшее развитие аппаратного обеспечения, алгоритмов шумоподавления и совершенствование инструментов разработки делают трассировку пути с использованием Vulkan все более привлекательным подходом для достижения нового уровня визуального качества в компьютерной графике реального времени. Дальнейшие исследования могут быть направлены на интеграцию более сложных моделей материалов, гибридные подходы (сочетание растеризации и трассировки) и адаптивные методы сэмплинга.

1. Andersson, E. Real-Time Path Tracing using Vulkan Ray Tracing. Master's Thesis, Linköping University, Department of Computer and Information Science. - 2021 [Электронный ресурс]. – Режим доступа: <http://liu.diva-portal.org/smash/get/diva2:1572197/FULLTEXT01.pdf> – Дата доступа: 30.03.2025.
2. Vulkan Ray Tracing Specification. Khronos Group. [Электронный ресурс]. – Режим доступа: https://registry.khronos.org/vulkan/specs/1.3-extensions/man/html/VK_KHR_ray_tracing_pipeline.html – Дата доступа: 30.03.2025.
3. Kajiya, J. T. The rendering equation // ACM SIGGRAPH Computer Graphics. – 1986. – Vol. 20. – №. 4. – Pp. 143-150.

Акименко Александр Владимирович, студент кафедры программного обеспечения и информационных технологий БГУИР, alexander0aki@gmail.com.

Научный руководитель: Красковский Павел Николаевич, старший преподаватель кафедры программного обеспечения и информационных технологий БГУИР, kraskovskij@bsuir.by.

СРАВНЕНИЕ ГРАФИЧЕСКОГО РЕНДЕРИНГА НА CPU, CUDA И RT-ЯДРАХ

В статье описывается понятие графического рендеринга, его техник, проводится сравнение рендера на CPU, CUDA и RT-ядрах.

ВВЕДЕНИЕ

С развитием технологий визуализации появляются всё более продвинутые способы рендеринга — от классического CPU до ускоренного CUDA и специализированных RT-ядер. Однако оборудование, способное использовать все эти подходы, остаётся дорогостоящим и не всегда доступным. В таких условиях особенно актуальным становится вопрос: действительно ли современные методы, такие как CUDA и RT-рендеринг, дают ощутимые преимущества по сравнению с CPU, и в каких задачах это оправдано? Сравнение этих трёх подходов помогает понять, насколько оправдана инвестиция в ту или иную технологию и в каких сценариях каждый метод показывает наилучший результат.

I. РЕНДЕРИНГ И ВИДЫ ТЕХНОЛОГИЙ, КОТОРЫЕ УЧАСТВУЮТ В ОБРАБОТКЕ СЦЕН

Рендеринг (визуализация) — это преобразование 3D-сцены в 2D-изображение с учётом параметров: геометрии объектов, освещения, камеры и материалов [1].

CPU-ядра — это мощные вычислительные единицы, выполняющие сложные логические, арифметические и управленческие операции. В рендеринге они используются для высокоточного оффлайн-рендеринга, где важна точность, например, в рендерах Arnold, Corona, V-Ray CPU, LuxCoreRender. Каждый пиксель или луч может обрабатываться отдельным потоком, что позволяет добиться фотореалистичного результата [2].

CUDA (Compute Unified Device Architecture) — это проприетарная архитектура параллельных вычислений от NVIDIA, позволяющая использовать ядра GPU для выполнения общих вычислений, в том числе задач рендеринга. В отличие от CPU, CUDA-ядра ориентированы на массивные параллельные потоки, где каждая задача, например, расчет цвета пикселя может выполняться независимо от других [3].

RT-ядра (Ray Tracing Cores) — это специализированные аппаратные блоки внутри GPU, например, NVIDIA RTX-серии, предназначенные исключительно для трассировки лучей. В отличие от универсальных CUDA-ядер, RT-ядра реализуют аппаратную поддержку основных операций трассировки — пересечений лучей с объектами сцены и построения иерархий ускорения [4].

II. РАБОТА РЕНДЕРИНГА НА CPU, CUDA И RT-ЯДРАХ НА АППАРАТНОМ УРОВНЕ

Рендеринг на CPU использует последовательную или умеренно параллельную обработку, так как CPU ориентирован на выполнение универсальных инструкций. Несмотря на меньшее число ядер по сравнению с GPU, каждое из них мощнее и подходит для сложных задач, включая ветвления, работу с памятью и математические вычисления. Это делает CPU оптимальным для оффлайн-рендеринга в архитектуре и визуальных эффектах. Рендереры на CPU (Arnold, Corona, V-Ray CPU) обрабатывают сцену (геометрию, материалы, свет, камеры) в ОЗУ, затем ядра CPU выполняют трассировку лучей и расчёт освещённости для пикселей или тайлов. Эффективность достигается за счёт кэшей L1–L3 и переиспользования данных, особенно в глобальном освещении [2].

CUDA-рендеринг применяется в Octane Render, Redshift, Iray для трассировки лучей, освещения, теней, отражений и постобработки. CUDA задействует все ресурсы GPU: регистры, ALU, кэши и специализированные блоки. Это обеспечивает высокую производительность без потери качества, особенно в режиме реального времени. Алгоритм работы:

- хост (CPU) подготавливает данные сцены и отправляет их в память GPU;
 - Запускаются CUDA-ядра, в которых каждый поток обрабатывает пиксель/треугольник/луч;
 - CUDA-ядра выполняют операции: арифметика с векторами, пересечения лучей и примитивов, шейдинг, вычисление освещённости;
 - результаты возвращаются в CPU или сохраняются на GPU для дальнейшей постобработки.
- В этом процессе CPU выступает только как координатор, а вычисления выполняются CUDA-ядрами. Также используются текстурные блоки и кэши для ускорения доступа к данным [5]. RT-ядра занимаются исключительно трассировкой, освобождая CUDA-ядра для других задач. Это повышает общую эффективность. Алгоритм работы следующий:
- CPU готовит сцену и передаёт на GPU;
 - CUDA-ядра запускают рендер-ядро и распределяют задачи;
 - RT-ядра берут на себя трассировку: вычисляют пересечения лучей с геометрией, возвращают результат в CUDA/шейдерные блоки;
 - CUDA-ядра и тензорные ядра выполняют шейдинг, постобработку, Denoising и так далее;
 - результат отсылается обратно в память и отображается [6].

III. ПРАКТИЧЕСКОЕ СРАВНЕНИЕ РЕНДЕРИНГОВ

Для практического сравнения рендеринга использовалась программа Blender с загруженной в неё тяжёлой для обработки сцены с несколькими автомобилями. На моделях применены блестящие материалы на кузове, а на стёклах есть отражения. Свет в сцене выставлен так, чтобы отражения лучей затрагивали сразу обе машины и отражались между объектами. Тестирование для всех типов ядер производилось в одинаковых условиях с использованием рендера Cycles в разрешении 1920x1080 пикселей при количестве samples 1225. Для проведения тестов использовался персональный компьютер со следующими характеристиками:

- CPU — 13th Gen Intel(R) Core(TM) i7-13700KF;
- GPU — ASUS TUF Gaming GeForce RTX 4070 Ti 12GB;
- RAM — 32 GB DDR5 6400 МГц.

Итоги проведённого сравнения приведены в таблице 1.

Таблица 1 – Результаты сравнения рендерингов

Тип ядер	Время рендеринга	Затраченная ОП
CPU	6 мин 28 сек	174,32 Мб
CUDA	37 сек	626,42 Мб
RT	21 сек	550,61 Мб

Как видно из результатов тестирования рендеринг на CPU самый долгий (в 10,5 раз дольше, чем на CUDA и RT ядрах), но при этом он затрачивает наименьшее количество оперативной памяти, чем другие рендеринги (3,6 раза меньше). Из всех методов рендеринга рендер на RT-ядрах оказался самым быстрым (в 18 раз быстрее CPU рендеринга и в 1,8 раза быстрее CUDA рендеринга) при этом он затрачивает меньше оперативной памяти чем CUDA рендеринг, но больше, чем CPU рендеринг (в 3,2 раза), однако скорость обработки явно окупается так как итоговый кадр выглядит одинаково на каждом виде рендеринга.

IV. ВЫВОДЫ

Результаты тестирования показали, что рендеринг на CPU занимает значительно больше времени, чем на CUDA и RT-ядрах, при минимальном использовании оперативной памяти. CUDA-рендеринг

обеспечивает заметный прирост скорости, но требует больше ресурсов, а наилучшую производительность продемонстрировали RT-ядра, сократив время рендера до минимума при умеренном потреблении памяти. Это подтверждает, что современные аппаратные ускорители действительно дают ощутимое преимущество в скорости и эффективности, особенно в ресурсоёмких сценах.

1. До конца загрузки осталось 19 часов: разбираем, что такое рендеринг? [Электронный ресурс]. – Режим доступа: <https://practicum.yandex.ru/blog/что-такое-rendering/> – Дата доступа: 10.04.2025.
2. Что такое ядро процессора и какие функции оно выполняет: подробное руководство [Электронный ресурс]. – Режим доступа: <https://mksegment.ru/a/что-такое-ядро-процессора-i-kakie-funkcii-ono-vypolnyaet?> – Дата доступа: 10.04.2025.
3. Best Hardware for GPU Rendering in Octane – Redshift – Vray [Электронный ресурс]. – Режим доступа: <https://www.cgdirector.com/best-hardware-for-gpu-rendering-in-octane-redshift-vray/> – Дата доступа: 10.04.2025.
4. Real-Time Ray Tracing [Электронный ресурс]. – Режим доступа: https://developer.nvidia.com/rtx/ray-tracing?sortBy=developer_learning_library%2Fsort%2Ftitle%3Aasc – Дата доступа: 10.04.2025.
5. Ray Tracing on a GPU with CUDA – Comparative Study of Three Algorithms [Электронный ресурс]. – Режим доступа: <https://gowtham-maran.github.io/Ray%20Tracing%20on%20a%20GPU%20with%20CUDA.pdf> – Дата доступа: 10.04.2025.
6. RT-ядра (Ray Tracing Cores) [Электронный ресурс]. – Режим доступа: <https://dzen.ru/b/Z5mgYL0KJ3qyR-0p> – Дата доступа: 10.04.2025.

Адамович Богдан Константинович, студент 3 курса кафедры вычислительных методов и программирования БГУИР, jemojem@yandex.ru

Научный руководитель: Кукин Дмитрий Петрович, Заведующий кафедры вычислительных методов и программирования БГУИР, kukin@bsuir.by.

МАТЕМАТИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ПРОГРЕССИИ ПЕРСОНАЖА

Исследуются математические модели прогрессии игровых персонажей, включая кривые мощности и системы обратной связи. Анализируются подходы к балансировке и их влияние на игровой опыт.

ВВЕДЕНИЕ

Прогрессия персонажей как ключевой элемент игрового дизайна требует моделирования кривых мощности и механизмов обратной связи. Рост сложности виртуальных миров ставит под вопрос эффективность традиционных методов балансировки. Цель работы — систематизация математических моделей прогрессии и разработка рекомендаций для их адаптивного применения в современных игровых системах.

I. КРИВЫЕ МОЩНОСТИ И ИХ КЛАССИФИКАЦИЯ

Кривые мощности формализуются через функцию $P(t)$, отражающую эффективность персонажа в момент времени t . В зависимости от характера изменения общей мощности системы выделяют три типа:

Положительная сумма ($\int \frac{dP}{dt} > 0$) характеризуется неограниченным ростом мощности, что типично для RPG и MMO-игр. Например, в World of Warcraft прокачка персонажа позволяет бесконечно увеличивать уровень и характеристики, создавая иллюзию постоянного прогресса. Однако такая модель приводит к инфляции контента: старые зоны и задания теряют актуальность, а разработчики вынуждены постоянно расширять игровой мир.

Нулевая сумма ($\int \frac{dP}{dt} = 0$) предполагает перераспределение мощности между участниками без общего роста. Этот подход доминирует в соревновательных играх, таких как Dota 2, где усиление одного героя компенсируется ослаблением другого. Недостатком является риск стагнации: игроки могут ощущать отсутствие долгосрочных целей.

Отрицательная сумма ($\int \frac{dP}{dt} < 0$) реализуется в survival-играх (например, Rust), где ресурсы истощаются, а постройки разрушаются со временем. Такие системы создают напряженную атмосферу, но провоцируют демотивацию из-за неизбежной потери прогресса.

II. МЕХАНИЗМЫ ОБРАТНОЙ СВЯЗИ

Обратные связи регулируют скорость изменения мощности и играют критическую роль в поддержании баланса:

Положительная обратная связь выражается как $\frac{dP_i}{dt} = kP_i(1 + \alpha(P_i - \bar{P}))$ и усиливает преимущество лидирующих игроков. В стратегиях типа Civilization развитые города генерируют больше ресурсов, ускоряя экспансию. Однако это приводит к дисбалансу: игроки, получившие раннее преимущество, доминируют без сопротивления.

Отрицательная обратная связь, формализуемая как $\frac{dP_i}{dt} = kP_i(1 - \beta(P_i - \bar{P}))$, компенсирует отставание. В гоночных играх (Mario Kart) отстающие получают мощные бонусы, что увеличивает шансы на победу. Недостаток — искусственное выравнивание, снижающее значимость мастерства.

III. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ И БАЛАНСИРОВКА

Современные игры редко используют чистые типы сумм или обратных связей. Например, в MMORPG Black Desert Online PvE-прогрессия строится на положительной сумме, а PvP-механики — на нулевой. Динамическая балансировка достигается через адаптивные алгоритмы, изменяющие параметры α и β в зависимости от стадии игры.

Важную роль играет матрица взаимодействий $A = [a_{ij}]$, где a_{ij} отражает эффективность игрока i против j . Условие баланса: $\forall i \exists j |a_{ij} - a_{ji}| < \epsilon$, где ϵ — допустимая погрешность. В MOBA-играх это реализуется через периодические патчи, корректирующие характеристики героев.

В современных играх часто комбинируют типы сумм на разных режимах. Например, в PvEvP-играх PvE-активности дают положительную сумму, PvP-столкновения работают по нулевой сумме, потеря ресурсов в PvP добавляет отрицательную сумму. Математически это выражается через кусочно-заданную функцию:

$$\frac{dP}{dt} = \begin{cases} k_1 P & \text{в PvE} \\ -k_2 P & \text{в PvP} \\ 0 & \text{в нейтральных зонах} \end{cases}$$

IV. РЕЗУЛЬТАТЫ И ОБСУЖДЕНИЕ

Анализ данных игровой аналитики показывает, что гибридные системы демонстрируют наибольшую устойчивость. Кривые мощности, сочетающие логистический рост $P(t) = \frac{P_{max}}{1 + e^{-k(t-t_0)}}$ со стохастической составляющей $\epsilon(t)$, обеспечивают предсказуемость при сохранении вариативности. Механизмы обратной связи, динамически подстраивающиеся под уровень игрока, снижают риск доминирования отдельных стратегий.

V. ВЫВОДЫ

Кривые мощности должны соответствовать целям игры: в PvE обеспечивать положительную сумму для ощущения прогрессии, а в PvP — стремиться к нулевой, чтобы сохранить баланс.

Комбинируя разные виды обратной связи, можно усилить ранние достижения игрока, но при

этом избежать эффекта "снежного кома когда преимущества накапливаются слишком быстро.

Персонализация прогрессии на основе игрового поведения — важное направление для будущих исследований. Адаптивные системы, учитывающие стиль игры, могут повысить вовлеченность и сделать опыт более увлекательным.

1. Smith, H. (2022). Game Balance Mathematics. Cambridge, MA: MIT Press. DOI: 10.1007/978-3-030-12345-6
2. Adams, E. (2021). Dynamic Systems in Game Design. Boca Raton, FL: CRC Press. ISBN: 978-1-032-34567-8
3. Hunicke, R. (2019). Formal Methods for Fair Play. Berlin: Springer. DOI: 10.1007/978-3-030-98765-4
4. Romero, B., & Schreiber, I. (2021). Игровой баланс. Точная наука геймдизайна [Game Balance: The Exact Science of Game Design]. Москва: Питер. ISBN: 978-5-4461-5432-1
5. Von Neumann, J., & Morgenstern, O. (1944). Theory of Games and Economic Behavior. Princeton, NJ: Princeton University Press.

Авдеева Татьяна Андреевна, студент кафедры вычислительных методов и программирования БГУ-ИР, tatushaken@gmail.com.

Научный руководитель: Гуревич Ольга Викторовна, ассистент, старший преподаватель, o.gurevich@bsuir.by

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ФИЛЬТРОВ НА ДАЛЬТНИЗМ

Рассматривается реализация алгоритма адаптации изображений для людей с нарушениями цветовосприятия. Производится сравнительный анализ существующих методов и их недостатков. Описано математическое представление алгоритма и его алгоритмика.

ВВЕДЕНИЕ

Одним из ключевых направлений в компьютерной графике и обработке изображений является обеспечение доступности визуального контента для людей с дальтонизмом. По статистике, около 4,5% населения страдает различными формами цветовой слепоты, что создаёт проблемы в медицине, образовании и цифровых интерфейсах.

Алгоритмы коррекции цветов позволяют преобразовать изображения так, чтобы они оставались информативными и эстетически приемлемыми для дальтоников. Наиболее распространённые методы включают:

- Матричные преобразования
- Алгоритмы локальной коррекции
- Нейросетевые подходы

Данная работа посвящена анализу и улучшению существующих алгоритмов.

I. АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДОВ И ИХ НЕДОСТАТКОВ

Матричные преобразования, такие как метод Бреттеля-Виена в пространстве LMS или линейные преобразования в RGB, применяют глобальную коррекцию цветов по фиксированным правилам. Например, протанопия моделируется умножением RGB-значений на матрицу 3×3 , что приводит к сдвигу красного спектра в оранжево-жёлтую область. Однако такие методы создают артефакты.

Алгоритмы локальной коррекции, такие как адаптивные LUT-таблицы и метод Хаффа, сталкиваются с принципиальным ограничением, связанным с дискретизацией цветового пространства. Они работают с заранее заданными шаблонами коррекции, которые не учитывают контекст изображения. Например, LUT-таблицы жёстко перераспределяют цвета в заранее определённые оттенки, что может приводить к неестественным скачкам в градиентах или потере важных деталей.

Нейросетевые методы, такие как GAN-архитектуры или свёрточные сети, показывают лучшие результаты. Например, модель CVD-Transformer на базе Vision Transformer сохраняет плавные градиенты на спутниковых снимках Landsat, но требует 1.2 ГБ видеопамати и 800 мс на обработку изображения 1024×1024 . Для сравнения, классический метод Vienot (1999) тратит всего 5 мс на аналогичной задаче, но искажает цветовые

соотношения в сложных сценах, таких как мультиспектральные изображения.

Таким образом, ни один из существующих методов не решает всех проблем: матричные преобразования слишком грубые, локальные алгоритмы недостаточно точны, а нейросети требуют непрактичных вычислительных ресурсов.

II. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

Алгоритм коррекции изображений для людей с дальтонизмом основан на последовательном анализе и преобразовании цветов с учётом особенностей зрительного восприятия. Процесс начинается с сегментации изображения, где нейросетевая модель разделяет контент на значимые элементы и фон. Это позволяет дифференцированно обрабатывать ключевые объекты и фоновые области.

На следующем этапе применяется кластеризация цветов для выявления доминирующих оттенков. Алгоритм группирует схожие пиксели, что помогает определить цветовую палитру изображения. Затем анализируются пары цветов, которые могут восприниматься неразлично при определённых типах дальтонизма, например, красно-зелёных нарушениях. Для этого используется колориметрическая система CIE Lab, которая позволяет количественно оценить различимость оттенков через метрику E, отражающую воспринимаемую разницу между цветами.

Коррекция выполняется адаптивно: фоновые тона плавно смещаются в сторону более различных оттенков, сохраняя естественность градиентов, а ключевые элементы (текст, объекты) заменяются на контрастные цвета, выбираемые из доступной для восприятия палитры. Например, красный может трансформироваться в оранжевый, а зелёный — в сине-зелёный.

Финальная проверка включает симуляцию дальтонизма через математические модели (например, Бреттеля) и оценку контрастности по стандарту WCAG, гарантируя, что скорректированное изображение остаётся функциональным и эстетически приемлемым для всех пользователей.

III. МАТЕМАТИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ

В основе алгоритма лежит математический аппарат, преобразующий цвета через последовательность точно определённых операций. Исходное изображение в RGB-пространстве сначала переводится в модель Lab через нелинейное преобразование, где L отвечает за светлоту, а компоненты a и

b задают положение цвета в хроматической плоскости.

Для измерения различий между цветами применяется усовершенствованная формула:

$$\Delta E_{00} = \sqrt{\left(\frac{\Delta L}{k_L S_L}\right)^2 + \left(\frac{\Delta C}{k_C S_C}\right)^2 + \left(\frac{\Delta H}{k_H S_H}\right)^2} + R_T \left(\frac{\Delta C}{k_C S_C}\right) \left(\frac{\Delta H}{k_H S_H}\right) \quad (1)$$

где ΔL , ΔC , ΔH - различия по светлоте, хроме и тону с корректировками, S_L , S_C , S_H - весовые коэффициенты восприятия, k_L , k_C , k_H - параметры условий наблюдения (по умолчанию 1), R_T - коэффициент вращения для синих тонов.

Расчёт контрастности строится на взвешенной сумме цветовых каналов, где коэффициенты 0.2126, 0.7152 и 0.0722 отражают вклад красного, зелёного и синего в воспринимаемую яркость соответственно. Финальная формула контраста:

$$\text{Hue} = \frac{L_1 + 0.05}{L_2 + 0.05} \quad (2)$$

где относительная яркость $L = 0.2126 \times R + 0.7152 \times G + 0.0722 \times B$ представляет собой отношение яркостей с добавлением слагаемого 0.05 в числитель и знаменатель, что предотвращает неопределённость при работе с тёмными цветами и обеспечивает устойчивость вычислений.

Баранова Янина Сергеевна, Рутковская Дарья Андреевна, студентки кафедры вычислительных методов и программирования БГУИР, ybaranova022@gmail.com, rutkovskayadaria@gmail.com.

Научный руководитель: Гуревич Ольга Викторовна, старший преподаватель, o.gurevich@bsuir.by.

IV. Выводы

Предлагаемый метод сочетает анализ изображения, адаптивную коррекцию и проверку различимости, обеспечивая лучшую читаемость для дальтоников. В перспективе это открывает путь к созданию гибридных систем, где нейросети используются лишь для первичного анализа контента, а коррекция цветов выполняется детерминированными алгоритмами – лучше сочетая точность машинного обучения с интерпретируемостью и надёжностью классической колориметрии.

Таким образом, метод не только решает актуальную проблему доступности визуального контента, но и задаёт направление для разработки более прозрачных и адаптируемых инструментов обработки изображений.

1. Brettel H., Viénot F., Mollon J.D. Computerized simulation of color appearance for dichromats // Journal of the Optical Society of America A, 1997, 2647–2655.
2. Jefferson, L., & Harvey, R. An interface to support color blind computer users. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06), 2006, 1535–1540.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ ГЛОБАЛЬНОГО ОСВЕЩЕНИЯ (PATH TRACING, PHOTON MAPPING, V-RAY)

В статье рассматриваются три модели глобального освещения - Path Tracing, Photon Mapping и V-Ray с использованием Brute Force и Light Cache. Проводиться сравнительный анализ их математических основ.

ВВЕДЕНИЕ

Глобальное освещение — ключ к физически корректному рендерингу. Важно выбрать подходящую модель.

I. ГЛОБАЛЬНОЕ ОСВЕЩЕНИЕ

Оно описывает полное световое взаимодействие (отражение, преломление, каустика) между поверхностями и основано на интегральном уравнении рендеринга:

$$L_o(x, \omega_o) = L_e(x, \omega_o) + \int_{\Omega} f_r(x, \omega_i, \omega_o) \cdot L_i(x, \omega_i) \cdot (\omega_i \cdot \mathbf{n}) d\omega_i$$

которое описывает вклад входящего света L_i .

II. PATH TRACING

Path Tracing реализует численное интегрирование уравнения рендеринга методом Монте-Карло, где считает только интеграл, а L_e в отдельном коде, когда путь попал в светящийся объект. Случайно выбираются направления к источникам отражения, формула итерации:

$$L_o(x, \omega_o) \approx \frac{1}{N} \sum_{i=1}^N \frac{L_i f_r \cos \theta}{p(\omega_i)}$$

Используется *importance sampling*: энергия зависит от угла падения света ($\cos \theta$). При выборе направлений пропорционально $\cos \theta$, формула упрощается:

$$\frac{f_r L_i \cos \theta}{\cos \theta / \pi} = \pi f_r L_i$$

что позволяет ускорить вычисления при сохранении физической корректности.

III. PHOTON MAPPING

Photon Mapping — двухфазный biased-алгоритм. Сначала поток фотонов трассируется и

сохраняются в карту. При попадании поток сохраняется и обновляется. Чтобы избежать рекурсии, применяется метод Russian Roulette. Затем энергия собирается в точке обзора по ближайшим фотонам в заданном радиусе, и поток вычисляется по формуле:

$$\Phi_e(x_0, \omega_0) = \frac{L_e(x_0, \omega_0) \cdot |\cos \theta|}{p(x_0, \omega_0) \cdot p_l \cdot N},$$

где $p(x_0, \omega_0)$ — плотность направления, p_l — плотность источника, а N — число фотонов.

IV. V-RAY

V-Ray — коммерческий рендер, использующий трассировку лучей и численное интегрирование. Здесь рассматриваются Brute Force и Light Cache. Первый использует метод Монте-Карло, как и path tracing. Второй — фотоподобный метод (без трассировки фотонов). Приближённая формула:

$$L_o(x) \approx \sum_{j=1}^N \frac{1}{N} f_r(x, \omega_j, \omega_0) L_i(x, \omega_j) (\omega_j \cdot \mathbf{n}),$$

позволяет оценить выходное излучение на основе суммы вкладов от выборки направлений.

V. ВЫВОДЫ

V-Ray - универсальный метод за счет использования разных алгоритмов получается быстро и качественно. Path Tracing наиболее точен, но самый медленный за счет численного интегрирования уравнения рендеринга методом Монте-Карло. Photon mapping трассирует лучи и лучше справляется с отражающими поверхностями.

1. Фapp, М. Physically Based Rendering: From Theory to Implementation / М. Фapp, В. Якоб, Г. Хамфрис. — Morgan Kaufmann, 2016. — 1200 с.

Бондарук Алина Андреевна, студент 3 курса факультета информационных технологий и управления БГУИР, maiko.taya@gmail.com.

Харланчук Диана Сергеевна, студент 3 курса факультета информационных технологий и управления БГУИР, Diana-harl@mail.ru.

Научный руководитель: Шатилова Ольга Олеговна, старший преподаватель кафедры вычислительных методов и программирования БГУИР, o.shatilova@bsuir.by

ЭНТРОПИЯ КАК МЕТРИКА ОЦЕНКИ ПОЛИГОНАЛЬНЫХ СЕТОК

В работе исследуется возможность применения энтропийного подхода для количественной оценки геометрической сложности полигональных 3D-моделей.

ВВЕДЕНИЕ

Современные методы анализа и оптимизации полигональных моделей требуют точных и универсальных метрик, способных учитывать как топологические, так и геометрические особенности формы. Одним из перспективных направлений является использование энтропии, понимаемой не в вероятностном, а в структурном смысле – как меры разнообразия состояний элементов сетки.

I. ПОНЯТИЕ ЭНТРОПИИ ПОЛИГОНАЛЬНОЙ СЕТКИ

В рамках доклада следует понимать энтропию в её неклассическом определении, как некую величину, показывающую, насколько много в системе различных состояний, но не отражающую того, насколько они устойчивы, то есть не связывающуюся с понятием беспорядка [1]. Такая трактовка предполагает, что энтропия указывает на размер множества всевозможных внутренних реализаций, неразличимых или эквивалентных с точки зрения внешнего наблюдателя.

Полигональная сетка, в свою очередь, может быть рассмотрена как дискретное представление поверхности или формы, определяемое совокупностью вершин, рёбер и граней, упорядоченных согласно определённой топологии. Понятие энтропии полигональной сетки, развиваемое в этом контексте, апеллирует к множеству всех возможных конфигураций, которые данная форма может принимать без нарушения её макроскопической целостности – включая объем, границы, и топологические свойства.

Таким образом, энтропия полигональной сетки есть мера её топологической и метрической гибкости при фиксированной форме. Она выражает, сколь многие внутренние устройства могут быть отнесены к одной и той же внешней репрезентации.

II. МЕТОДИКА РАСЧЁТА ЭНТРОПИИ ПОЛИГОНАЛЬНОЙ СЕТКИ

Под системой, описываемой энтропией, в данном случае понимается совокупность ориентированных поверхностей (граней) трёхмерной модели, каждая из которых характеризуется собственной

нормалью – единичным вектором, перпендикулярным данной поверхности.

Ввиду непрерывной природы нормалей и невозможности напрямую оперировать бесконечным числом возможных направлений, вводится процедура дискретизации. Каждая нормаль нормализуется и округляется до заданного шага по каждой координате. Это позволяет разделить непрерывное пространство направлений на конечное множество областей, каждая из которых трактуется как отдельное состояние системы.

На следующем этапе определяется мощность множества уникальных направлений, возникающих в результате дискретизации. В соответствии с выбранной трактовкой, энтропия в данном случае представлена не как величина, зависящая от вероятностного распределения, а как мера структурной вариативности – количество различных состояний, в которых могут находиться элементы модели в рамках заданного допущения о точности. Именно это число и принимается в качестве энтропийной оценки, отражающей геометрическую сложность формы. Чем больше уникальных направлений выявлено, тем более структурно насыщенной оказывается модель, и тем выше её энтропия.

III. ВЫВОДЫ

Предложенная методика расчёта энтропии на основе анализа нормалей демонстрирует способность фиксировать геометрическое разнообразие модели независимо от числа полигонов. Энтропия отражает структурную насыщенность формы, а не плотность сетки. Такой подход позволяет надёжно дифференцировать модели, схожие по топологии, но различные по характеру поверхности, и может быть полезен в автоматизированных системах анализа, оптимизации и классификации трёхмерных данных.

1. Чепкасов, В. Л. Новые смыслы понятия энтропии, или к вопросу о неклассическом варианте понятия энтропии / В. Л. Чепкасов, Т. Л. Михайлова // Международный журнал экспериментального образования. – 2014. – № 6-1. – С. 164-167.

Бубен Всеволод Дмитриевич, студент кафедры вычислительных методов и программирования БГУ-ИР, usevaladbuben@gmail.com.

Научный руководитель: Кужин Дмитрий Петрович, заведующий кафедрой вычислительных методов и программирования БГУИР, кандидат технических наук, доцент, kugin@bsuir.by.

ВЛИЯНИЕ ПАРАМЕТРОВ СИСТЕМ ЧАСТИЦ НА ВРЕМЯ РЕНДЕРИНГА В 3DS MAX

Рассматриваются зависимости времени рендеринга от параметров систем частиц в редакторе трёхмерной графики 3ds Max. Исследуется влияние количества частиц, скорости движения и вариантов отображения частиц.

ВВЕДЕНИЕ

Одной из ключевых задач трёхмерной графики является достижение реалистичного визуального эффекта при оптимальном времени рендеринга. Визуальные эффекты, построенные на системах частиц (Particle Systems), активно применяются при моделировании природных явлений. Однако увеличение реалистичности часто приводит к росту времени обработки сцены. Цель данной работы — оценить, как различные параметры систем частиц влияют на время рендеринга в редакторе трёхмерной графики Autodesk 3ds Max [1].

I. ПАРАМЕТРЫ СИСТЕМ ЧАСТИЦ

В 3ds Max представлены различные типы эмиттеров частиц, предоставляющий более гибкое управление. Основными параметрами, влияющими на производительность, являются:

- количество частиц;
- размер частиц;
- тип геометрии;
- наличие модификаторов (Wind, Gravity, Deflector и др.).

Для исследования была выбрана сцена с Sprau, в которой варьировались параметры количества частиц от 1000 до 500000, скорости от 30 до 180 единиц и варианты отображения частиц: Drops, Dots и Ticks.

II. ЭКСПЕРИМЕНТАЛЬНЫЕ ДАННЫЕ

Тестирование проводилось на ПК с процессором AMD Ryzen 5 5500U и оперативной памятью 8,00 ГБ. Использовался встроенный рендерер Arnold. Ниже приведены результаты эксперимента (рис. 1).

Бузенкова Ксения Геннадьевна, студент 3 курса факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники, ksenia.buzenkova@mail.ru,

Кривцов Ярослав Геннадьевич, студент 3 курса факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники, trydny2010@gmail.com.

Научный руководитель: Кужин Дмитрий Петрович, заведующий кафедрой вычислительных методов и программирования Белорусского государственного университета информатики и радиоэлектроники, кандидат технических наук, доцент, kukin@bsuir.by.

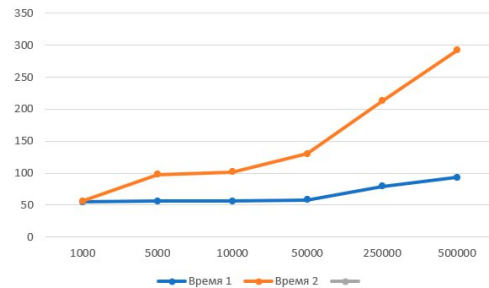


Рис. 1 – Влияние количества частиц на время рендеринга

На графике по вертикальной оси отложено время рендеринга, а по горизонтальной — количество частиц. Синяя линия отражает зависимость скорости рендеринга от количества частиц, а оранжевая — зависимость времени рендеринга от скорости движения частиц.

III. ВЫВОДЫ

Результаты эксперимента показывают, что основным фактором, влияющим на производительность при рендеринге сцен с системами частиц в 3ds Max, является количество частиц и скорость движения частиц. Разные варианты отображения частиц никак не повлияли на время рендеринга сцены. Применение невысоких скоростей падения частиц и оптимизация количества частиц позволяют существенно сократить время рендеринга без значительной потери качества.

1. Toxigon: Mastering Render Times in 3ds Max for Faster Workflows [Электронный ресурс]. Режим доступа: https://toxigon.com/improving-render-times-in-3ds-maxgoogle_vignette

СЕЗОННОСТЬ В ИГРОВОЙ ИНДУСТРИИ СНГ: АНАЛИЗ ПРОДАЖ И ПОТРЕБИТЕЛЬСКОГО СПРОСА

Рассматривается влияние сезонности на активность и доходность в игровой индустрии СНГ. Устанавливается зависимость DAU и Revenue от праздничных периодов, дней недели и времени суток.

ВВЕДЕНИЕ

Для анализа влияния сезонности анализируются данные ежедневной активности пользователей и доходности игр. В качестве объектов исследования выбраны AAA проекты, такие как «Metro Exodus», «Atomic Heart», «Loop Hero», и мобильная игра жанра «три в ряд». Для выявления закономерностей используется сопоставление пиков и спадов метрик DAU (Daily Active Users) и Revenue с календарными событиями.

I. АНАЛИЗ ЕЖЕДНЕВНОЙ АКТИВНОСТИ ПОЛЬЗОВАТЕЛЕЙ В AAA ИГРАХ

Пиковая активность пользователей соответствует релизным событиям, праздничным периодам и распродажам. Игра «Atomic Heart» демонстрирует высокую чувствительность к маркетинговым кампаниям и обновлениям контента, «Metro Exodus» — умеренные, но предсказуемые колебания, связанные с праздниками и традициями повторного прохождения, в то время как «Loop Hero» сохраняет стабильную нишевую активность с реакцией на фестивальные события (см.рис.1.).

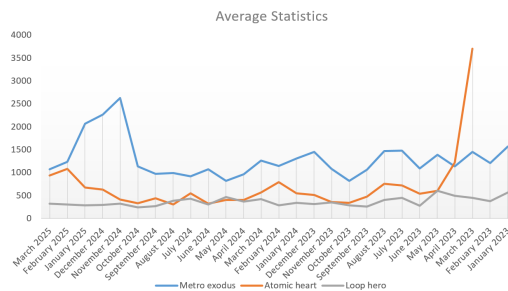


Рис. 1 – График изменения метрики DAU

II. АНАЛИЗ МЕТРИК В КАЗУАЛЬНЫХ ИГРАХ

Сезонность проявляется в характерных подъёмах вовлечённости в январе и декабре, обусловленных новогодними каникулами и праздничными акциями. Повышенные показатели DAU и Revenue также наблюдаются в марте и мае на фоне календарных праздников. В летний период фиксируются минимальные значения активности и доходов, что

связано с отпускным временем и хорошими погодными условиями. В сентябре спад постепенно прекращается, пользователи возвращаются к рабочему ритму. Осень характеризуется постепенным восстановлением интереса, пиковые значения выручки приходятся на декабрь.

III. ВЛИЯНИЕ СЕЗОННОСТИ НА АКТИВНОСТЬ В ТЕЧЕНИЕ НЕДЕЛИ

Установлено, что наибольшая вовлечённость и выручка фиксируются в выходные дни, особенно в воскресенье, а также в вечерние часы с 20:00 до 22:00. Будние дни демонстрируют стабильную, но сниженную активность, с минимальными значениями в понедельник. На фоне общего суточного ритма наибольшая платёжная активность приходится на вечер, тогда как утренние и ночные часы характеризуются низкой выручкой.

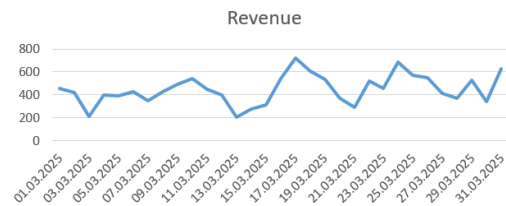


Рис. 2 – График изменения метрики Revenue

IV. ВЫВОДЫ

Эффективное планирование рекламных кампаний, акций, внутриигровых событий и запусков контента с учётом сезонных и поведенческих паттернов способно значительно повысить коммерческую эффективность игрового продукта и обеспечить более глубокую вовлечённость аудитории.

Список литературы

1. Сабиров, В. К. Игра в цифры. Как аналитика позволяет видеоиграм жить лучше / В. К. Сабиров – 2021. – С. 238-245.
2. Steam Charts [Электронный ресурс] / Steamcharts.com. – Режим доступа: <https://steamcharts.com> – Дата доступа: 14.04.2025.
3. Devtodev – аналитика мобильных приложений [Электронный ресурс] / Devtodev.com. – Режим доступа: <https://www.devtodev.com> – Дата доступа: 14.04.2025.

Бурая Александра Константиновна, студент 3 курса факультета информационных технологий и управления БГУИР, sandra.k.buraya@gmail.com.

Научный руководитель: Гуревич Ольга Викторовна, старший преподаватель кафедры вычислительных методов и программирования БГУИР, o.gurevich@bsuir.by.

МАТЕМАТИЧЕСКИЕ ПРИНЦИПЫ И АЛГОРИТМЫ МОРФИНГА В 3DS MAX

В научной работе исследуются математические принципы и алгоритмы морфинга в 3ds Max, включая линейную и взвешенную интерполяцию. Работа объясняет, как математические методы обеспечивают реалистичность и качество плавных трансформаций объектов.

ВВЕДЕНИЕ

В современной 3D-графике и анимации одной из актуальных проблем является создание плавных и реалистичных трансформаций объектов, таких как лицевые выражения персонажей или изменение форм сложных моделей. Эти задачи особенно важны в индустрии кино, видеоигр и виртуальной реальности, где качественная анимация напрямую влияет на восприятие контента. Одним из эффективных решений данной проблемы является технология морфинга – процесса плавного преобразования одного объекта в другой. В 3ds Max морфинг реализуется через модификатор Morpher [1], который использует математические методы интерполяции.

I. МАТЕМАТИЧЕСКИЕ ОСНОВЫ МОРФИНГА

Морфинг основан на следующих математических концепциях:

- линейная интерполяция;
- взвешенная интерполяция.

II. ЛИНЕЙНАЯ ИНТЕРПОЛЯЦИЯ

Линейная интерполяция – это математический метод, используемый для создания плавного перехода между двумя объектами или состояниями [2]. В контексте морфинга в 3ds Max линейная интерполяция применяется для плавного преобразования одного объекта в другой. Основная формула линейной интерполяции между двумя объектами O_0 и O_1 выглядит следующим образом:

$$O(t) = (1 - t) \times O_0 + t \times O_1.$$

где O_0 – исходный объект, форма которого будет изменяться; O_1 – целевой объект; $t \in [0; 1]$ – это параметр, который определяет степень преобразования исходного объекта в целевой; $O(t)$ – результирующий объект.

Этот процесс повторяется для всех вершин объекта, что обеспечивает плавное преобразование всей формы.

Голенчук Михаил Викторович, студент 3 курса факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники, mgolenchuk@gmail.com.

Научный руководитель: Кукин Дмитрий Петрович, заведующий кафедрой вычислительных методов и программирования БГУИР, кандидат технических наук, доцент, kukin@bsuir.by.

III. ВЗВЕШЕННАЯ ИНТЕРПОЛЯЦИЯ В МОРФИНГЕ

Взвешенная интерполяция – это расширение линейной интерполяции, которое позволяет учитывать влияние нескольких целевых объектов на итоговую форму [3]. В 3ds Max эта техника реализована через модификатор Morpher, где каждый целевой объект может быть назначен на отдельный канал с определенным весом. Основная формула взвешенной интерполяции выглядит следующим образом:

$$\begin{cases} O(t) = (1 - \sum_{i=1}^n w_i) \cdot O_0 + \sum_{i=1}^n w_i \cdot O_i, & \text{если } \sum_{i=1}^n w_i \leq 1, \\ O(t) = \sum_{i=1}^n w_i \cdot O_i, & \text{если } \sum_{i=1}^n w_i > 1. \end{cases}$$

где O_0 – исходный объект; O_i – целевые объекты; w_i – это вес i -го канала, который определяет степень влияния целевого объекта на итоговую форму; $O(t)$ – результирующий объект.

IV. ВЫВОДЫ

Линейная и взвешенная интерполяция являются ключевыми методами в морфинге, позволяя создавать плавные и реалистичные трансформации объектов. Линейная интерполяция проста и эффективна для задач с двумя формами, а взвешенная интерполяция предоставляет гибкость и контроль для работы с несколькими целевыми объектами.

Список литературы

1. Autodesk Knowledge Network. The Morpher Modifier in 3ds Max [Electronic resource]. – 2025. – Mode of access: <https://knowledge.autodesk.com>. – Date of access: 26.03.2025.
2. Parent, R. Computer Animation: Algorithms and Techniques / R. Parent. – 3rd ed. – Morgan Kaufmann, 2012. – P. 454–456.
3. Lengyel, E. Mathematics for 3D Game Programming and Computer Graphics / E. Lengyel. – 3rd ed. – Cengage Learning, 2012. – 545 p.

ПРИМЕНЕНИЕ МАШИННОГО ОБУЧЕНИЯ ДЛЯ АВТОМАТИЗАЦИИ СОЗДАНИЯ ТРЕХМЕРНЫХ АНИМАЦИЙ ИЗУЧЕНИЕ ВОЗМОЖНОСТЕЙ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ГЕНЕРАЦИИ АНИМАЦИЙ ИЛИ СОЗДАНИЯ РЕАЛИСТИЧНЫХ ДВИЖЕНИЙ

Статья рассматривает различные аспекты использования нейронных сетей для создания 3D-анимаций

ВВЕДЕНИЕ

3D-анимация играет ключевую роль в современных цифровых индустриях — от игр до киноиндустрии. Машинное обучение, особенно методы глубокого и усиленного обучения, открывают новые подходы, позволяя автоматизировать ключевые этапы создания анимации: генерацию движений, стилизацию, мимику, рендеринг, тем самым ускорить производство.

I. АРХИТЕКТУРА АВТОМАТИЗИРОВАННЫХ СИСТЕМ

Системы автоматизированной 3D-анимации на базе ИИ включают модули для обработки входных данных, генерации движений, стилизации, визуализации сцен, а также взаимодействия с художником, позволяя ему вносить правки на любом этапе. Такая система функционирует в интерактивном или полностью автономном режиме. Подобную модульную структуру предложили Xue Lu и Bo Liu, используя САД-технологии в сочетании с глубоким обучением для создания автоматических систем генерации 3D-анимации с функциями адаптации и интеллектуальной синхронизации движений и выражений.

II. АЛГОРИТМИЧЕСКАЯ ОСНОВА

Одним из базовых алгоритмов в задаче генерации движения является autoregressive decoder-only трансформер. Модель, вдохновлённая архитектурой GPT, предсказывает следующую скелетную позу по предыдущим, обучаясь на реалистичных траекториях и адаптируясь к нужному стилю движения. Математически процесс предсказания описывается функцией, которая получает все предыдущие кадры и выдаёт следующий: $\hat{x}_{t+1} = \text{Decoder}(x_1, x_2, \dots, x_t)$. Качество предсказания оце-

нивается с помощью функции потерь, которая измеряет среднее отклонение сгенерированной позы от эталонной: $\mathcal{L} = \frac{1}{n} \sum_{i=1}^n \|x_i - \hat{x}_i\|_1$.

III. ГЕНЕРАЦИЯ АССЕТОВ И СЦЕН

Для создания фонов и окружения используются диффузионные модели, такие как Stable Diffusion. Они поэтапно восстанавливают изображение из шума, учитывая визуальный контекст и условия генерации, например текстовое описание сцены. По сути, модель обучается постепенно устранять шум, приближаясь к заданному изображению. Это формализуется через стохастическое обновление состояния: $x_{t-1} = \mu_\theta(x_t, t, c) + \sigma_t \cdot \varepsilon$, где $\varepsilon \sim \mathcal{N}(0, I)$.

IV. ВЫВОДЫ

Машинное обучение, в частности нейросетевые и генеративные методы, показали высокую эффективность в автоматизации 3D-анимации. Комбинация предобученных моделей, интерактивных компонентов и адаптивных алгоритмов оптимизации позволяет создать гибкие системы, сокращающие время и трудозатраты при сохранении качества. В перспективе основное внимание будет уделено интеграции многомодальных источников (текст, речь, движение), генеративному контролю и более глубокому взаимодействию с пользователем.

1. XLu X., Liu B. Automatic Generation and Design of 3D Animation Based on Deep Learning Technology // Computer-Aided Design Applications. – 2025. – Vol. 22, S1. – P. 46–59.
2. Gao Q. Design and Implementation of 3D Animation Data Processing Development Platform Based on Artificial Intelligence // Computational Intelligence and Neuroscience. – 2022. – Article ID 1518331, 7 p. – DOI: 10.1155/2022/1518331.

Голодко Анна Андреевна, студент 3 курса факультета информационных технологий и управления БГУИР, kamennyugol@gmail.com.

Сахацкая Яна Руслановна, студент 3 курса факультета информационных технологий и управления БГУИР, yanusya.js@gmail.com.

Научный руководитель: Гуревич Ольга Викторовна, старший преподаватель кафедры вычислительных методов и программирования БГУИР, o.gurevich@bsuir.by.

АЛГОРИТМ ГЕНЕРАЦИИ ЛАНДШАФТА UNITY

Рассматриваются принцип работы Terrain, основные алгоритмы с помощью которых генерируется ландшафт.

ВВЕДЕНИЕ

Система Terrain представляет собой набор инструментов для создания и формирования окружающей среды в компьютерной графике. Она сочетает в себе элементы ручного редактирования и процедурной генерации, что позволяет эффективно создавать масштабные и детализированные ландшафты. Одной из ключевых составляющих terrain-систем является использование алгоритмов шумов и процедурных моделей, обеспечивающих разнообразие форм рельефа, реалистичность и управляемость. Такие алгоритмы позволяют задавать высоты точек на сетке, имитировать горные хребты, холмы, равнины, каньоны и другие природные структуры.

I. ШУМ ПЕРЛИНА

Шум Перлина - это метод генерации градиентного шума. Он используется в компьютерной графике для создания реалистичных текстур, таких как облака, дым, вода, ландшафты и многое другое. В отличие от обычного белого шума, Перлин-шум обладает «плавностью» и непрерывностью, что делает его пригодным для создания органичных и естественных структур. Шум Перлина часто применяется в генерации высотных карт и процедурных текстур. Метод основан на интерполяции случайных векторов, распределённых по сетке, что обеспечивает мягкие переходы между значениями. Также он может быть использован в 2D, 3D и даже n-мерных пространствах.

II. DIAMOND-SQUARE

Алгоритм Diamond-Square используется для генерации фрактальных ландшафтов. Он получил широкое распространение в играх и симуляторах благодаря своей простоте и способности создавать разнообразные и реалистичные рельефы. Метод начинается с квадратной сетки, где значения задаются в углах, а затем итеративно заполняются по алгоритму: сначала центр квадрата (diamond), затем середины сторон (square). Алгоритм повторяется на всё более мелких участках, при этом добавляется случайный шум с уменьшением амплитуды. Такой подход позволяет достичь характерной фрактальной структуры ландшафта.

III. ВОРОНОЙ

Диаграмма Вороного - метод пространственного разбиения, основанный на расстоянии до заданных точек. Каждая точка «владеет» областью пространства, состоящей из всех точек, которые бли-

же к ней, чем к любой другой. Такой подход широко применяется в генерации биомов, городов, текстур и карт. В графике диаграммы Вороного позволяют создавать интересные и естественные паттерны, напоминающие клетки, кристаллы, карты рек и трещины. Существует множество вариаций метода: взвешенные, ориентированные и стохастические.

IV. ФРАКТАЛЬНЫЙ ШУМ

Фрактальный шум - это комбинация нескольких слоёв базового шума (например, Перлина), наложенных друг на друга с различными частотами и амплитудами. Такой подход позволяет создавать более детализированные и реалистичные текстуры по сравнению с обычным шумом. Часто используется для имитации природных объектов: гор, облаков, ландшафтов. Фрактальный шум обычно реализуется через технику fBm (fractal Brownian motion), в которой каждый последующий слой вносит всё меньший вклад в итоговое значение, создавая богатую и сложную структуру.

V. ПРИМЕНЕНИЕ АЛГОРИТМОВ В СИСТЕМЕ TERRAIN

Система Terrain представляет собой платформу для создания реалистичных ландшафтов с использованием процедурных алгоритмов. Она позволяет комбинировать различные методы генерации шума, такие как Шум Перлина, Diamond-Square, диаграммы Вороного и фрактальный шум, чтобы формировать разнообразные элементы окружающей среды — от мягких холмов до резких горных хребтов. Например, Шум Перлина используется в Terrain для моделирования плавных переходов высот, что позволяет создавать реалистичные холмы, равнины и дюны. Он особенно эффективен при создании больших территорий с природными формами рельефа. Алгоритм Diamond-Square применяется для генерации карт высот благодаря своей способности создавать детализированные и естественные формы рельефа с переменным уровнем грубости. Этот метод хорошо подходит для моделирования горных ландшафтов и пересечённой местности. Диаграммы Вороного в системе Terrain часто используются для имитации образования островов, долин или равномерного распределения объектов, таких как скалы или кратеры. Они помогают разбивать территорию на сегменты, придавая ландшафту структуру и ритм. Фрактальный шум в Terrain позволяет сочетать шумы с разными частотами и амплитудами, создавая многослойные, детализированные ландшафты. Он особенно полезен для симуляции эрозии, высотных перепадов и микрострукту-

ры поверхности. С помощью Terrain разработчики могут настраивать параметры генерации — частоту шума, масштаб, высотные пороги и интенсивность сглаживания — что делает этот инструмент гибким и мощным для создания сложных и разнообразных виртуальных миров.

VI. Вывод

Сравнительный анализ алгоритмов процедурной генерации демонстрирует, что каждая из технологий обладает своими уникальными особенностями и областью применения. Шум Перлина отличается высокой плавностью и реалистичностью, что делает его идеальным для текстур и мягких переходов. В то же время фрактальный шум позволяет добиться гораздо более высокой детализации за счёт наложения множественных частот. Алгоритм Diamond-Square, благодаря своей простоте и эффективности, широко используется для генерации рельефов и ландшафтов, обеспечивая характерную фрактальную структуру. Диаграммы Вороного, напротив, находят применение там, где требуется чёткое разбиение пространства на зоны — будь то симуляция природных узоров или генерация территорий. Таким образом, выбор подходящего алгоритма зависит от поставленных задач. Система Terrain

объединяет эти алгоритмы, предоставляя удобную платформу для их комбинированного применения. Это позволяет создавать реалистичные и визуально богатые ландшафты с высокой степенью гибкости. Возможность настраивать параметры генерации и комбинировать различные методы делает Terrain мощным инструментом в области процедурной генерации окружающей среды.

Список литературы

1. Ken Perlin (1985). An Image Synthesizer. ACM SIGGRAPH Computer Graphics, 19(3), 287–296.
2. Ebert, D., Musgrave, F., Peachey, D., Perlin, K., Worley, S. (2003). Texturing Modeling: A Procedural Approach. 3rd edition, Morgan Kaufmann.
3. Okabe, A., Boots, B., Sugihara, K., Chiu, S. N. (2000). Spatial Tessellations: Concepts and Applications of Voronoi Diagrams. 2nd ed., John Wiley Sons.
4. Lagae, A., Dutré, P. (2008). A Comparison of Methods for Procedural Noise Functions. Computer Graphics Forum, 27(1), 114–129.
5. Fournier, A., Fussell, D., Carpenter, L. (1982). Computer Rendering of Stochastic Models. Communications of the ACM, 25(6), 371–384.
6. Musgrave, F.K., Kolb, C.E., Mace, R.S. (1989). The Synthesis and Rendering of Eroded Fractal Terrains. ACM SIGGRAPH Computer Graphics, 23(3), 41–50.
7. Gustavson, S. (2005). Simplex Noise Demystified.
8. Inigo Quilez – Fractal Brownian Motion (fBm)

Галушко Артём Вячеславович, студент 3 курса Факультета информационных технологий и управления БГУИР, artem.galushko.04@mail.ru

Научный руководитель: Гуревич Ольга Викторовна, старший преподаватель кафедры вычислительных методов и программирования БГУИР, o.gurevich@bsuir.by.

МЕТОДЫ РЕТОПОЛОГИИ ПОЛИГОНАЛЬНОЙ СЕТКИ

В работе приводится описание методов ретопологии полигональной сетки, разработанных на основе алгоритмов оптимизации геометрии в программе Blender.

ВВЕДЕНИЕ

В настоящее время при создании 3D-объектов часто используют скульптинг, при помощи которого создается высокополигональная модель. Для ее облегчения без потери производительности применяется ретопология.

I. ОСНОВНЫЕ ПОДХОДЫ РЕТОПОЛОГИИ

Ретопология – это процесс создания сетки на основе существующей высокополигональной модели. Главная цель – сохранить внешний вид оригинальной модели, но при этом сократить количество полигонов.

Существуют три основных подхода к ретопологии. Ручная ретопология трудоемкая, но самая точная. Сетка создается вручную, следуя форме исходной модели. Автоматическая ретопология осуществляется с помощью алгоритмов, которые перестраивают сетку. Комбинированная ретопология делается автоматически, а потом ее результат дорабатывается вручную [1].

II. REMESH-РЕТОПОЛОГИЯ В BLENDER

В Blender для автоматической ретопологии можно использовать модификатор Remesh, который перестраивает геометрию объекта, создавая новую сетку на основе формы модели. Он предлагает четыре основных способа обработки сетки: Blocks, Smooth, Sharp и Voxel [2]. На примере ретопологии высокополигональной модели, состоящей из 500000 полигонов проанализируем, какой метод быстрее, какой лучше сохраняет исходную форму объекта.

Киселёва Татьяна Сергеевна, студентка 3 курса факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники, tanyshakiseliova@gmail.com. .

Шеремет Ольга Анатольевна, студентка 3 курса факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники, noname09901234@gmail.com.

Научный руководитель: Кукин Дмитрий Петрович, заведующий кафедры вычислительных методов и программирования Белорусского государственного университета информатики и радиоэлектроники, кандидат технических наук, доцент, kudin@bsuir.by

Метод Blocks, разбивая объект на блоки, создает простую кубическую полигональную сетку, максимально сокращает полигоны. Однако он плохо сохраняет форму, подходит для грубого упрощения. Метод Blocks оставил 3000 полигонов из 500000 в исходной модели за 2 секунды, но почти потерял форму модели.

Метод Smooth создаёт сглаженную сетку, нет резких переходов между полигонами. Создание сетки заняло 6 секунд, осталось 50000 полигонов, мелкие детали потерялись, но форма модели сохранена.

Метод Sharp сохраняет резкие грани модели и прекрасно сохраняет углы. На создание сетки понадобилось 7 секунд, а полигонов осталось 48000.

Метод Voxel преобразует объект в воксельную форму и затем пересчитывает сетку, лучше всего сохраняет детализацию, но требует больших ресурсов. На создание сетки понадобилось 22 секунды, а количество полигонов стало 123000.

ЗАКЛЮЧЕНИЕ

Voxel Remesh – самый ресурсоёмкий, Blocks подходит для быстрой оптимизации, без сохранения точной формы. Smooth и Sharp оптимальны, так как позволяют сохранить форму исходной модели при небольших потерях детализации.

Список литературы

1. GarageFarm.NET, "Retopology: A Comprehensive Guide to 3D Model Optimization"[Электронный ресурс], <https://clck.ru/3LPwhN>.
2. Blender 3.6 Manual, "Ретопология"[Электронный ресурс], <https://clck.ru/3LQYUL>.

СОЗДАНИЕ АНИМАЦИИ ЛИЦА

Рассматривается создание анимации лица для разных сфер с помощью разных технологий, и дальнейшее развитие и инновации создания анимации.

ВВЕДЕНИЕ

Лицевая анимация объединяет компьютерную графику, биомеханику, психологию восприятия и ИИ, эволюционировав от простого морфинга до нейросетей, воспроизводящих тончайшие мимические нюансы. Применение: Кино (CG-персонажи) Игры (The Last of Us, Cyberpunk 2077) VR/метавселенные Мультфильмы и реклама (Pixar, Disney) Технологии лицевой анимации востребованы благодаря запросу на реализм в цифровых медиа.

I. ОСНОВНЫЕ МЕТОДЫ СОЗДАНИЯ ЛИЦЕВОЙ АНИМАЦИИ

1.2. Motion Capture (MoCap) Принцип работы: Маркерный/безмаркерный трекинг мимики. Запись движений камерами (4-120 шт.). Перенос данных на 3D-модель. Технологии: Маркерные: Vicon, OptiTrack (высокая точность). Безмаркерные: Faceware, TrueDepth (удобство). Применение: "Аватар" Планета обезьян Hellblade, LA Noire 1.3. Ключевая анимация Ручная расстановка ключевых кадров. Использует blendshapes и костные системы. Применение: мультфильмы Pixar/DreamWorks, Overwatch, Fortnite. 1.4. Процедурная анимация Автогенерация мимики по: Голосу (JALI, Paragayo). Эмоциям (Unreal Engine, Maya). Примеры: NVIDIA Audio2Face, DeepMotion 1.5. ИИ-анимация Нейросетевой анализ и воссоздание мимики. Технологии: Deepfake, MetaHuman (Unreal Engine 5)

II. ЭТАПЫ СОЗДАНИЯ ЛИЦЕВОЙ АНИМАЦИИ

2.1. Подготовка модели Моделирование: Оптимизированная топология (особенно глаза, рот). Работа с полигональной сеткой. Риггинг: Настройка FACS для мимики. Создание виземов и скиннинга. 2.2. Производство анимации Захват данных: Калибровка актера. Запись и обработка сырых данных. Очистка и ретаргетинг: Удаление артефактов. Коррекция трекинга. Анимация: Базовые эмоции и синхронизация губ. Детализация (микродвижения, динамика). 2.3. Финальная обработка Оптимизация: Настройка для real-time. Подготовка LOD-уровней. Рендеринг: Кино: Arnold, RenderMan. Игры: Unreal Engine, Unity.

III. ИНСТРУМЕНТЫ И СОФТ

Инструменты для лицевой анимации 3D-моделирование: Blender - бесплатный инструмент для создания топологии и скульптинга мимики (поддержка blend shapes). Maya - профессиональное моделирование с акцентом на анимационную

топологию. ZBrush - лучший инструмент для детальной анатомической проработки. Риггинг: Maya - продвинутая настройка FACS и скелетных систем. Mixamo - автоматический риггинг с готовыми решениями. Motion Capture: Faceware - безмаркерный захват для инди-проектов. Vicon - профессиональный маркерный захват (точность 0.1 мм). iClone - доступный захват через веб-камеру. Анимация: Character Animator - live-анимация с веб-камеры. Cascadeur - физически корректная анимация с ИИ. ИИ-решения: Omniverse - генерация анимации по аудио. DeepMotion - нейросетевой анализ видео. Движки: Unreal 5 - MetaHuman для фотореализма. Unity - оптимален для мобильных и VR.

IV. ОСНОВНЫЕ ПРИНЦИПЫ РЕАЛИСТИЧНОЙ ЛИЦЕВОЙ АНИМАЦИИ

Преувеличение - усиление эмоций для лучшей читаемости (например, более широкая улыбка) Тайминг - контроль скорости: быстрые движения для удивления, медленные - для грусти Сглаживание - плавные переходы между выражениями Эти принципы создают баланс между выразительностью и реализмом.

V. РИГГИНГ В ЛИЦЕВОЙ АНИМАЦИИ: ПРИНЦИПЫ И МЕТОДЫ

Суть риггинга: Создание управляющей системы для анимации мимики, выступающей виртуальным аналогом мышечной системы. Ключевые компоненты: Скелетная система: Виртуальные кости с анатомической иерархией. Контроллеры для основных зон лица. Система ограничений и зависимостей. Blend Shapes: 50-300 предустановленных выражений. Комбинации для сложных эмоций. Процесс создания: Анализ анатомии и мимики. Разработка системы управления: Многоуровневые контроллеры. Параметрическое управление. Механизмы симметрии. Тестирование и оптимизация. Характеристики: 200-500 управляющих элементов. Срок настройки: 2-8 недель. Современные решения (MetaHuman) сочетают скелетную анимацию и blend shapes. Критерии качества: Естественные деформации. Интуитивное управление. Гибкость настройки. Оптимизированная производительность.

VI. ПЕРСПЕКТИВНЫЕ НАУЧНЫЕ ИННОВАЦИИ В СОЗДАНИИ ЛИЦЕВОЙ АНИМАЦИИ

1. Нейроинтерфейсы (BCI): Прямой захват нервных импульсов (Neuralink, Kernel). Анализ эмоций через ЭЭГ/fNIRS. Точность до 95%. Голографический захват: Лазерная интерферометрия (0.01 мм точность). Квантовые сенсоры (1000 fps). 360°

сканирование без маркеров. 3. Генеративный ИИ: Мультимодальные системы (аудио+текст+эмоции). Мгновенное создание аватаров по фото. Автокоррекция артефактов. 4. Квантовые технологии: Молекулярное моделирование кожи. Реал-тайм рендеринг сложных эффектов. Оптимизация для высокополигональных моделей. 5. Бионические интерфейсы: Тактильные контроллеры с обратной связью. Голографические рабочие станции. Коллаборация в смешанной реальности. Преимущества: Ускорение производства на 60-80%. Вызовы: Этические вопросы цифровых двойников. Энергоэффективность решений. Стандартизация и обучение.

VII. Выводы

Лицевая анимация прошла значительный путь развития — от примитивных морфинговых преобразований до сложных нейросетевых систем, способных воспроизводить тончайшие нюансы человеческой мимики. В данной работе были рассмотрены ключевые методы создания лицевой анимации, включая Motion Capture, ключевую анимацию, процедурные технологии и машинное обучение, а также этапы производства, инструментарий и принципы достижения реализма. Современные технологии, такие как метавселенные, нейроадаптивные системы и генеративный ИИ, открывают новые горизонты для индустрии. Однако наряду с техническим прогрессом возникают этические вызовы, связанные с использованием цифровых двойников, защитой персональных данных и преодолением эффекта «зловещей долины».

Козлов Артем Викторович, студент 3 курса Факультета информационных технологий и управления БГУИР, ar_koz@mail.ru.

Научный руководитель: Кукин Дмитрий Петрович, доцент кафедры вычислительных методов и программирования БГУИР, kukin@bsuir.by.

Перспективы развития лицевой анимации связаны с нейрокомпьютерными интерфейсами, квантовыми вычислениями и голографическим захватом движения, которые позволят добиться беспрецедентного уровня реализма и ускорят процесс производства. Внедрение инноваций потребует пересмотра стандартов качества, разработки новых протоколов безопасности и обучения специалистов работе с передовыми инструментами. В заключение можно констатировать, что лицевая анимация продолжает стремительно развиваться, стирая границы между реальным и виртуальным. Будущее этой области лежит в синергии технологий, искусства и науки, что откроет новые возможности для кинематографа, игровой индустрии, виртуальной реальности и коммуникации в цифровую эпоху.

Список литературы

1. Ekman, P. (1992). "Facial Expressions of Emotion: New Findings, New Questions". *Psychological Science*.
2. Magnenat-Thalmann, N., Thalmann, D. (2018). "Handbook of Virtual Humans". Wiley Publishing.
3. Williams, R. (2009). "The Animator's Survival Kit". Faber Faber.
4. Zhang, L., et al. (2020). "Deep Learning for 3D Face Animation". *ACM Computing Surveys*.
5. Bickel, B., et al. (2015). "Design and Fabrication of Materials with Desired Deformation Behavior". *ACM Transactions on Graphics*.
6. Faceware Technologies. (2022). "Markerless Facial Motion Capture Whitepaper".
7. NVIDIA. (2022). "Audio2Face: AI-Powered Facial Animation Solution".

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ В КОМПЬЮТЕРНЫХ ИГРАХ

Рассматривается влияние искусственного интеллекта в компьютерных играх на игровую индустрию, и дальнейшее развитие и инновации игр.

ВВЕДЕНИЕ

Искусственный интеллект (ИИ) — одна из ключевых технологий в современных компьютерных играх. Он отвечает за поведение неигровых персонажей (NPC), адаптацию сложности, генерацию контента и создание реалистичного игрового опыта. В этом докладе мы рассмотрим основные виды игрового ИИ, его применение и перспективы развития.

I. Роль ИИ в играх

Игровой ИИ решает несколько важных задач: 1. Управление NPC: ИИ контролирует поведение противников, союзников и нейтральных персонажей, обеспечивая их реалистичное взаимодействие с игроком и окружающим миром. 2. Планирование действий: ИИ отвечает за такие аспекты, как поиск пути и тактика в бою, что позволяет NPC принимать обоснованные решения в зависимости от ситуации. 3. Динамическая адаптация: ИИ может изменять уровень сложности игры в зависимости от навыков игрока, что делает игровой процесс более увлекательным и доступным. 4. Генерация контента: ИИ используется для создания процедурных миров и случайных квестов, что увеличивает реиграбельность и разнообразие игрового опыта. 5. Имитация человеческого поведения: В мультиплеерных играх ИИ может управлять ботами, которые ведут себя как реальные игроки, что позволяет заполнить пустоты в игре.

II. Основные виды игрового ИИ

Детерминированные системы ИИ 1. Конечные автоматы (FSM). Четко заданные состояния (атака/бегство/ожидание). Простая логика переходов. 2. Деревья поведения (Behavior Trees). Иерархия условий и действий. Пример: проверка дистанции → атака. Применение: The Last of Us, Dark Souls (Рис. 2). 3. Нечеткая логика. Вероятностные решения. Адаптация к угрозам. Используется: Total War, Civilization. Системы планирования 1. GOAP. Поиск оптимальных действий для цели. Реалистичное поведение (F.E.A.R.). 2. HTN. Разбивка задач на подзадачи. Сложное поведение (Shadow of Mordor). Машинное обучение 1. Нейросетевые боты. Адаптация к игроку. Примеры: Dota 2, StarCraft II. 2. Генетические алгоритмы. Эволюция поведения NPC. Применение: Creatures, Galactic Arms Race

III. Будущее искусственного интеллекта в игровой индустрии

Умные NPC с памятью и эмоциями. NPC будут запоминать действия игрока и менять отношение (например, торговец начнет избегать вора). Технологии вроде Inworld AI и NVIDIA ACE уже делают диалоги реалистичнее. Динамические миры. Животные будут охотиться и мигрировать, а города — жить своей жизнью. ИИ-помощник в разработке. Генерация осмысленных локаций (не просто руины, а замок с историей). Автоматизация артов, кода, озвучки (Unity Sentis, ChatGPT). Персонализированный геймплей Сюжет и враги адаптируются под стиль игрока (больше стражников для любителя стелса). ИИ-компаньоны смогут учиться и даже бунтовать. Новые жанры Игры, где игрок — ИИ, вирус или цифровой разум. ИИ-директор (как в Left 4 Dead), но анализирующий эмоции через камеру или микрофон. Проблемы Этические вопросы (вина за "смерть" NPC). Технические ограничения (требуются мощные ПК и интернет). ИИ сделает игры живыми, но разработчикам предстоит решить немало сложностей.

IV. Заключение

Игровой ИИ продолжает эволюционировать, переходя от простых скриптовых систем к сложным нейросетевым алгоритмам. В будущем игры станут более адаптивными, а виртуальные персонажи будут почти неотличимы от реальных людей. Это приведет к созданию умных врагов и союзников, динамических сюжетов без повторений и полной симуляции "живого" мира. Использование машинного обучения позволит ИИ анализировать поведение игроков и адаптироваться к их стилю игры. Враги смогут изучать стратегии игрока и подстраиваться под них, создавая уникальные тактики. ИИ также сможет генерировать контент в реальном времени, создавая новые квесты и диалоги на основе действий игрока, что сделает каждую сессию уникальной. Виртуальные персонажи будут развивать свои отношения с игроками, что добавит глубины взаимодействию. Игры смогут предлагать персонализированный опыт, учитывая предпочтения каждого игрока. В итоге, игровой ИИ станет ключом к созданию погружающих и увлекательных игр, которые оставят незабываемые впечатления.

Список литературы

1. Russell, S., Norvig, P. (2010). Artificial Intelligence: A Modern Approach. Prentice Hall.

2. Baker, C. (2014). Game AI Pro: Collected Wisdom of Game AI Professionals. CRC Press.
3. Millington, I., Funge, J. (2009). Artificial Intelligence for Games. Morgan Kaufmann.
4. Schaefer, J., Smith, R. (2015). Procedural Content Generation in Games. Springer.
5. Bates, J. (1994). "The Role of Emotion in Believable Agents". In Proceedings of the First International Conference on Autonomous Agents.
6. Aylett, R., Louchart, S. (2003). "Towards a Narrative Theory of Virtual Agents". In Proceedings of the 2003 International Conference on Intelligent Virtual Agents.
7. Kallmann, M., Thalmann, D. (2004). "Real-Time Motion Planning for Character Animation". In Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation.

Макаревич Андрей Сергеевич, студент 3 курса Факультета информационных технологий и управления БГУИР, 25makar2005@gmail.com.

Козлов Артем Викторович, студент 3 курса Факультета информационных технологий и управления БГУИР, arkoz@mail.ru.

Научный руководитель: Кукин Дмитрий Петрович, доцент кафедры вычислительных методов и программирования БГУИР, kukin@bsuir.by.

ИСПОЛЬЗОВАНИЕ ДИАГРАММ ВОРОНОГО И ТРИАНГУЛЯЦИИ ДЕЛОНЕ ДЛЯ СОЗДАНИЯ ТРЁХМЕРНЫХ ИГРОВЫХ ЛАНДШАФТОВ

Рассматривается применение диаграммы Вороного и триангуляции Делоне для процедурной генерации трёхмерных игровых ландшафтов.

ВВЕДЕНИЕ

Процедурная генерация ландшафтов с использованием диаграммы Вороного и триангуляции Делоне позволяет создавать структурно обоснованные игровые сцены с реалистичным рельефом. В работе описан подход к их применению и демонстрируется его реализация в среде Blender. Цель статьи — показать, как с помощью этих геометрических структур можно построить трёхмерную игровую сцену.

I. ДИАГРАММА ВОРОНОГО

Диаграмма Вороного конечного множества точек S на плоскости представляет такое разбиение плоскости, при котором каждая область этого разбиения образует множество точек, более близких к одному из элементов множества S , чем к любому другому элементу множества [1].

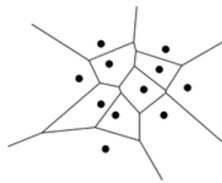


Рис. 1 – Диаграмма Вороного

Существуют различные способы построения диаграммы Вороного. Самым простейшим считается алгоритм, основанный на построении серединных перпендикуляров, его вычислительная сложность оценивается как $O(n^4)$. Однако оптимальным временем для построения диаграммы Вороного для множества из N точек считается $O(N \log N)$ с помощью алгоритма Форчуна [Алгоритм замечательной прямой для диаграмм Вороного, С. Форчун, 1986] или рекурсивного метода.

Область применения диаграммы Вороного весьма широка. В вычислительной геометрии она используется прежде всего для решения задачи близости точек, отсюда вытекает и её значимость в картографии и геолокации. В игровой индустрии, например, было найдено её применение как основы для системы навигации игрового движка [2].

II. ТРИАНГУЛЯЦИЯ ДЕЛОНЕ

Триангуляцией называется планарное разбиение плоскости на M фигур, из которых одна является внешней бесконечной, а остальные — треугольниками. Говорят, что триангуляция удовлетворяет

условию Делоне, если внутри окружности, описанной вокруг любого построенного треугольника, не попадает ни одна из заданных точек триангуляции. Такая триангуляция называется триангуляцией Делоне [3].

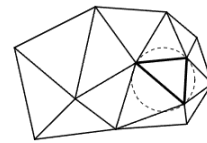


Рис. 2 – Триангуляция Делоне

Триангуляция Делоне и диаграмма Вороного являются двойственными структурами: каждая область или «ячейка» диаграммы соответствует вершине триангуляции Делоне. Если две ячейки соседствуют по стороне, то в триангуляции обязательно есть отрезок между двумя соответствующими точками [4].

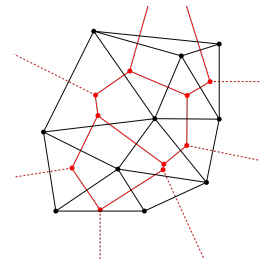


Рис. 3 – Связь диаграммы Вороного и триангуляции Делоне

Это свойство можно использовать для создания реалистичных игровых ландшафтов, например, использовать сетку Вороного как полигональную меш-основу, а с помощью графов Делоне генерировать реки, прокладывая дороги или долины [5].

III. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ГЕНЕРАЦИИ ИГРОВОГО ЛАНДШАФТА НА ОСНОВЕ ДИАГРАММЫ ВОРОНОГО

Рассмотрим задачу создания трёхмерного ландшафта на основе диаграммы Вороного. Для выполнения поставленной задачи была использована система трёхмерного моделирования и визуализации Blender, а также её встроенный инструмент процедурного моделирования Geometry Nodes. Выбор Blender обусловлен его широкими возможностями по работе с геометрическими структурами,

поддержкой нодового моделирования и наличием встроенной текстуры Voronoi, позволяющей легко реализовать визуальное разбиение пространства на полигоны.

1. Создание полигональной сетки на основе диаграммы Вороного.

На первом этапе создаётся двумерная полигональная сетка, соответствующая диаграмме Вороного. Для этого применяется узел Voronoi Texture, накладываемый на исходную плоскость. Визуализация текстуры позволяет получить уникальные границы ячеек, ассоциированных с семенными точками.

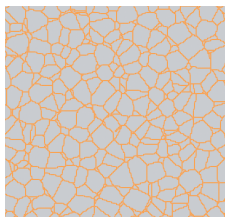


Рис. 4 – Наложение текстуры Вороного

2. Определение суши и воды.

Для разделения ландшафта на сушу и воду используется процедурная маска в форме окружности. Все точки, расположенные внутри окружности, считаются сушей, вне её — водой. С целью повышения реалистичности контура материка к форме маски добавляется шум Перлина, что позволяет получить неровные береговые линии.

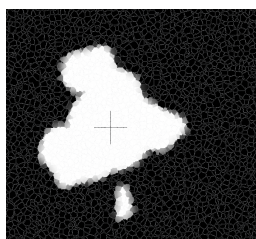


Рис. 5 – Выборка суши по окружности с шумом

3. Формирование рельефа.

На области, отнесённые к суше, накладывается дополнительный шум, модифицирующий координату по оси Z. Таким образом создаётся рельеф: возвышенности, низменности и другие топографические элементы.

Корытко Злата Сергеевна, студент 3 курса факультета информационных технологий и управления БГУИР, korytkozlata@mail.ru

Кушель Ксения Олеговна, студент 3 курса факультета информационных технологий и управления БГУИР, ksusha.kushel@gmail.com

Научный руководитель: Кукин Дмитрий Петрович, заведующий кафедрой вычислительных методов и программирования БГУИР, кандидат технических наук, доцент, kukin@bsuir.by.

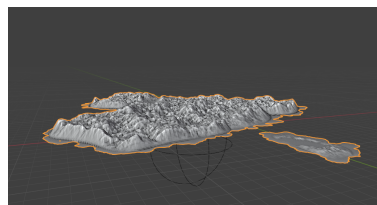


Рис. 6 – Генерация рельефа с использованием шума

4. Наложение цвета и текстур.

Финальный этап заключается в визуальном оформлении сцены. Полигоны, граничащие с водой, окрашены в жёлтый цвет (песчаный берег), внутренние области суши — в зелёный (растительность), а водные участки — в синий. Цветовая дифференциация облегчает восприятие структуры ландшафта.



Рис. 7 – Визуализация с наложением текстур и цвета

IV. Выводы

В статье представлен способ генерации трёхмерного ландшафта на основе диаграммы Вороного с последующим формированием рельефа, реализованный средствами Blender. Полученный результат подтверждает применимость описанного метода для визуального моделирования игрового пространства.

1. Препарта Ф., Шеймос М. Вычислительная геометрия: Введение. — М.: Мир, 1989. — С. 295.
2. Диаграмма Вороного и её применения [Электронный ресурс] — Режим доступа: <https://habr.com/ru/articles/309252/#Part3>
3. Скворцов А.В. Обзор алгоритмов построения триангуляции Делоне // Вычислительные методы и программирование. — 2002. — Т. 3, № 1. — С. 16.
4. Об одном интересном свойстве триангуляции Делоне [Электронный ресурс] — Режим доступа: <https://habr.com/ru/articles/828610/>
5. Polygonal Map Generation for Games [Электронный ресурс] — Режим доступа: <http://www-cs-students.stanford.edu/~amitp/game-programming/polygon-map-generation/>

КОМЬЮНИТИ-МЕНЕДЖМЕНТ И ЕГО РОЛЬ В МАРКЕТИНГОВОЙ СТРАТЕГИИ ИГРОВОГО ПРОДУКТА

В статье рассматривается понятие комьюнити-менеджмента как ключевого элемента в продвижении и развитии игр.

ВВЕДЕНИЕ

Современная игровая индустрия представляет собой высококонкурентную и стремительно развивающуюся сферу, в которой успех продукта зависит не только от его технической реализации или визуального исполнения, но и от эффективности коммуникации с аудиторией. На фоне обостряющейся борьбы за внимание пользователя становится очевидным, что одной из ключевых задач становится не столько привлечение новых игроков, сколько их дальнейшее удержание и формирование лояльности к бренду.

Одним из важнейших инструментов удержания аудитории является работа с сообществом. Цель данной статьи – обратить внимание на значимость управления взаимодействием с аудиторией в процессе продвижения и развития игры.

I. ЧТО ТАКОЕ КОМЬЮНИТИ-МЕНЕДЖМЕНТ

Комьюнити-менеджмент – это формирование и развитие добровольного сообщества людей, которые объединены общими целями и интересами [1]. Термин зародился как раз в игровой индустрии и изначально подразумевал взаимодействие разработчиков с игроками: техподдержку, публикацию новостей, ответы на вопросы игроков на форумах, «чистку» внутриигрового чата от флуда и др.

В современной гейм-индустрии основными задачами комьюнити-менеджера являются:

- полное ведение проектного веб-сайта (верстка новостей, гайдов, описания акций и т.д.);
- формирование контент-плана с дальнейшими публикациями информационного и развлекательного характера;
- мониторинг, модерация комментариев и ответы на них;
- составление отчетов с указанием результатов деятельности и реакции сообщества на действия компании [2].

Для эффективного взаимодействия с аудиторией комьюнити-менеджер использует широкий спектр цифровых каналов, включая социальные сети (такие как VK, Twitter, Instagram, TikTok), специализированные платформы (Discord, Reddit, Steam), официальные форумы, а также внутриигровые механизмы – опросы, системы обратной связи и общения внутри клиента. Эти инструменты позволяют не только оперативно реагировать на запросы

пользователей, но и проводить постоянный анализ удовлетворенности потребностей игроков.

В данном контексте важно понимать, что комьюнити-менеджер является не просто модератором или контент-мейкером, а полноценным связующим звеном между игроками и командой разработчиков. Он превращает эмоции и ожидания пользователей в основу для развития проекта и других важных стратегических решений.

II. КОМЬЮНИТИ-МЕНЕДЖМЕНТ И МАРКЕТИНГ

В наши дни на рынке вакансий граница между понятиями SMM-специалиста и комьюнити-менеджера сильно размыта. Зачастую один человек выполняет две эти роли одновременно, что логично, ведь они используют схожие каналы – социальные сети, мессенджеры, форумы – и работают с одной и той же аудиторией. Однако на практике такое объединение нередко приводит к перегрузке и смещению приоритетов в сторону более измеримых SMM-задач, тогда как работа с сообществом требует долгосрочного подхода и не всегда даёт немедленные результаты.

Хотя задачи SMM-специалиста и комьюнити-менеджера могут казаться схожими, их цели и подходы принципиально различаются. Основная задача специалиста по социальным медиа – продвижение продукта и увеличение охвата аудитории через контент и метрики эффективности, такие как просмотры, переходы и клики. Комьюнити-менеджмент, в свою очередь, сосредоточен на выстраивании устойчивых и долгосрочных отношений с пользователями, поддержке их вовлечённости и укреплении доверия к продукту и бренду.

Если первый опирается преимущественно на количественные показатели, то второй ориентирован на качественную обратную связь – анализирует комментарии, отслеживает настроения аудитории и помогает адаптировать продукт к её ожиданиям. В своей работе комьюнити-менеджер использует методы, связанные с психологией (понимание поведения, мотивации, реакций отдельных пользователей) и социологией (наблюдение за групповыми тенденциями, массовыми реакциями и формированием субкультур внутри сообщества).

Социологический подход позволяет структурировать сообщество, выявлять ключевые группы и налаживать эффективное взаимодействие с ними. Это отличает управление сообществом от маркетинга, который прежде всего ориентирован на формирование интереса к продукту и его презентацию. Однако эти направления не противопостав-

ляются, а, напротив, дополняют друг друга. Маркетологи фокусируются на привлечении внимания, тогда как комьюнити-менеджеры создают условия для его удержания. Таким образом, взаимодействие двух этих ролей обеспечивает комплексный подход к продвижению: от первой встречи пользователя с продуктом до формирования устойчивой лояльности.

Такое сотрудничество делает комьюнити-менеджмент важным элементом маркетинговой стратегии, сохраняя при этом его уникальную направленность на построение внутреннего диалога и доверия между компанией и её аудиторией.

III. ПРИМЕРЫ

Для более наглядного понимания значимости взаимодействия с аудиторией на различных этапах жизненного цикла продукта целесообразно обратиться к конкретным примерам игровых студий, добившихся как значительных успехов, так и столкнувшихся с неудачами в результате своей коммуникационной стратегии с сообществом.

Так, положительным примером эффективного комьюнити-менеджмента выступает студия *Larian Studios*, продемонстрировавшая высокий уровень открытости и обратной связи с аудиторией в период раннего доступа к игре *Baldur's Gate 3*. Разработчики активно включали пользователей в процесс разработки, в том числе путём публичного обсуждения вносимых изменений, основанных на отзывах игроков. Подобная стратегия способствовала формированию лояльного сообщества и высокому пользовательскому рейтингу на цифровых платформах.

В противоположность этому, компания *10:10 Games* продемонстрировала пример недостаточной работы с обратной связью, что негативно отразилось на релизе её продукта *Funko Fusion*. Несмотря на декларации открытости к предложениям игро-

ков, их замечания фактически не были учтены, что вызвало волну критических отзывов. Впоследствии это привело не только к финансовым потерям, но и к сокращению значительной части сотрудников студии. По сообщениям источников, руководство проекта стремилось ускорить завершение разработки, игнорируя при этом потребности и ожидания аудитории [3].

IV. ВЫВОДЫ

Комьюнити-менеджмент в игровой индустрии представляет собой неотъемлемую часть стратегического продвижения продукта и тесно связан с маркетинговыми задачами, хотя и не сводится к ним. В условиях высокой конкуренции удержание игрока, формирование лояльности и доверия становятся важнее однократного привлечения, и именно здесь ключевую роль играет эффективное взаимодействие с сообществом.

Комьюнити-менеджер выступает связующим звеном между разработчиком и аудиторией, обеспечивая постоянный обмен информацией, сбор обратной связи и адаптацию продукта к ожиданиям пользователей. Это требует не только технических навыков, но и глубинного понимания социальных процессов.

Список литературы

1. Комьюнити-менеджмент [Электронный ресурс]. Режим доступа: <https://www.unisender.com/ru/glossary/komyuniti-menedzhment/#anchor-1>, свободный.
2. Игровой комьюнити-менеджмент [Электронный ресурс]. – Режим доступа: https://stopgame.ru/blogs/topic/101139/igrovoy_komyuniti_menedzhment, свободный.
3. EXCLUSIVE: 10:10 Games Employees Offered No Severance, Layoffs Happened While On Holiday, And More [Электронный ресурс]. – Режим доступа: <https://insider-gaming.com/1010-games-employees-offered-no-severance-laid-off-while-on-holiday-and-more/>, свободный.

Кушель Ксения Олеговна, студентка 3 курса факультета информационных технологий и управления БГУИР, ksusha.kushel@gmail.com

Научный руководитель: Фролова Дарья Александровна, старший преподаватель кафедры экономики БГУИР, d.frolova@bsuir.by

РАЗРАБОТКА ЧАТ-БОТА ДЛЯ АВТОМАТИЗИРОВАННОГО ПОЛУЧЕНИЯ РАСПИСАНИЯ

Современные информационные системы вузов предоставляют данные о расписании через API, однако их использование требует автоматизации для удобства студентов. Разработан чат-бот на языке Python, использующий API ИИС «БГУИР: Университет» для получения актуального расписания и отправки его пользователям в мессенджере.

ВВЕДЕНИЕ

Обеспечение оперативного доступа студентов к информации об аудиториях, названиях дисциплин и ФИО преподавателей через мессенджеры представляет актуальную задачу. Для повышения удобства получения расписания предлагается разработка программного решения для упрощения получения расписания студентами.

I. АНАЛИЗ АКТУАЛЬНОСТИ

Оценка целесообразности внедрения чат-бота проведена методом анкетирования студентов, проживающих в четвёртом общежитии БГУИРа. Анализ актуальности показал, что 50 процентов респондентов не обладают информацией о расписании и аудиториях, что подчеркивает сложность своевременного доступа к данным. Вопросы анкеты касались частоты использования мессенджеров, их типов, источников получения расписания, а также готовности использовать чат-бот. Результаты свидетельствуют, что 51 процент опрошенных изъявили желанием иметь возможность посмотреть расписание и аудиторию через чат-бот(см.рис.1.)

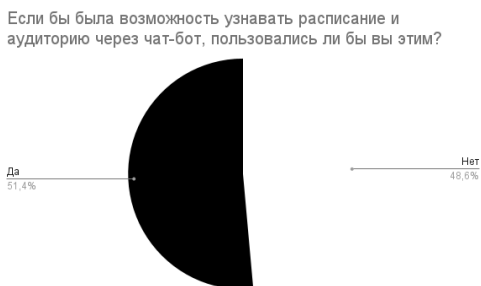


Рис. 1 – Результаты опроса

II. СОЗДАНИЕ ЧАТ-БОТА

Разработанный чат-бот создан на языке программирования Python с использованием набора библиотек для обеспечения функциональности.

Линкевич Алексей Олегович, студент специальности электронные системы и технологии ФКП БГУИР, leshalinkevich@gmail.com.

Научный руководитель: Панасик Алексей Александрович, ассистент кафедры вычислительных методов и программирования БГУИР, магистр технических наук, a.panasik@bsuir.by.

Для получения данных о расписании реализована интеграция с API ИИС (iis.bsuir.by), через которое извлекаются данные в формате JSON, после чего они обрабатываются для дальнейшего использования. Функционал включает предоставление расписания по запросу пользователя и автоматическую отправку уведомлений перед занятиями с указанием аудитории, названия дисциплины и ФИО преподавателя. В процессе создания применены библиотеки requests для выполнения HTTP-запросов к API, telebot и telebot.types (включая ReplyKeyboardMarkup, KeyboardButton) для взаимодействия с мессенджером Telegram, а также datetime и timedelta для работы с временными метками, time для управления задержками, threading для реализации многопоточной отправки уведомлений и logging для записи логов работы программы(см.рис.2.)

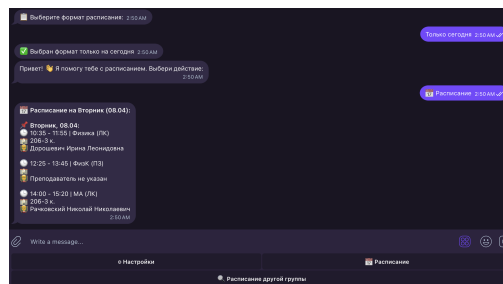


Рис. 2 – Пример работы чат-бота

III. ВЫВОДЫ

Разработанное решение способствует повышению доступности расписания для студентов. В дальнейшем возможно добавление новых функций.

- Документация API ИИС «БГУИР: Университет» [Электронный ресурс] – Режим доступа: <https://iis.bsuir.by/api/>. – Дата доступа: 20.04.2025.;
- Документация библиотеки PyTelegramBotAPI [Электронный ресурс] – Режим доступа: [https://pypi.org/project/pyTelegramBotAPI/..](https://pypi.org/project/pyTelegramBotAPI/) – Дата доступа: 20.04.2025.;

FRAMEWORK RPGBUILDER

Рассматривается фреймворк для Unity RPGBuilder.

I. ВВЕДЕНИЕ

RPGBuilder представляет собой специализированный фреймворк для Unity, предназначенный для упрощения разработки RPG-игр. В данном отчете представлены результаты исследования функциональных возможностей данного инструмента, его архитектуры и практического применения.

II. ОБЗОР ПЛАТФОРМЫ

Основные характеристики: -Интеграция с Unity (версии 2019.4 LTS и выше) -Визуальный редактор для конструирования RPG-элементов; -Поддержка скриптов на C#; -Модульная система компонентов;

Установка и настройка: - Требуется Unity Hub с установленной версией Unity; - Импорт пакета RPGBuilder через Package Manager; - Настройка проекта через конфигурационный файл.

III. ИННОВАЦИИ В ГЕЙМДИЗАЙНЕ

Система персонажей

Есть классы (воин, маг, лучник) и расы (эльф, орк, человек) с уникальными параметрами. Базовая статистика включает силу, ловкость, интеллект, выносливость — всё настраивается. Расовые бонусы: эльфы +10% к скорости, орки +15% к урону в ближнем бою.

Дерево навыков с визуальным редактором (перетаскивание нод). Типы навыков: активные (атаки, заклинания), пассивные (увеличение урона, регенерация), трансформации (временное усиление). Условия разблокировки: уровень, квесты, предметы.

Инвентарь с слотами экипировки (оружие, броня, аксессуары). Типы предметов: одноручные/двуручные мечи, шлемы, нагрудники, кольца. Настройка веса, сортировка по типу/редкости/уровню.

Прокачка: гибкая кривая опыта (XP), ручное/автоматическое распределение атрибутов при уровне, капы уровней.

Диалоговая система

Визуальный редактор с Drag & Drop для веток диалогов. Типы реплик: NPC, выбор игрока, триггеры (предметы/условия). Интеграция с квестами: старт заданий, изменение репутации с фракциями.

Квестовая система

Типы квестов: сбор (10 грибов), убийство (5 гоблинов), доставка (письмо), разговор (диалог с NPC). Награды: опыт, золото, предметы, локации. Условия: таймеры, скрытые цели.

Мир и локации

Генерация ландшафта: кисти для рельефа, префабы (деревья, камни), зоны спавна. Точки сохранения: костры (ручное), города (автосохранение).

Тип квеста	Описание	Пример
Сбор	Найти N предметов	"Принеси 10 грибов"
Убийство	Уничтожить врагов	"Убей 5 гоблинов"
Доставка	Отнести предмет	"Отнеси письмо в соседний город"
Разговор	Поговорить с NPC	"Узнай у кузнеца про меч"

Рисунок 1 - Типы квестов

IV. ПРЕИМУЩЕСТВА И НЕДОСТАТКИ

Преимущества: Экономия времени — не нужно писать системы с нуля (экономия 3-6 месяцев разработки). Готовые решения — инвентарь, диалоги, квесты уже реализованы. Визуальные редакторы — не нужно кодить базовые вещи. Поддержка сообщества — есть документация и готовые ассеты.

Недостатки: Ограниченная гибкость — если нужен нестандартный геймплей, придется переписывать код. Производительность — может тормозить на слабых устройствах из-за накладных расходов. Зависимость от фреймворка — если разработчик прекратит поддержку, придется переделывать.

V. ВЫВОДЫ

RPGBuilder представляет собой мощный инструмент для быстрой разработки RPG в Unity, особенно подходящий для инди-разработчиков и небольших студий. Хотя фреймворк имеет некоторые ограничения, его модульная архитектура и богатый функционал делают его отличным выбором для создания качественных RPG-проектов без необходимости разработки всех систем с нуля.

Критерий	RPGBuilder	Своя реализация
Скорость разработки	***** (быстро)	** (медленно)
Гибкость	** (ограничена)	***** (полная)
Поддержка	*** (есть документация)	* (всё на себе)
Производительность	*** (средняя)	***** (оптимизировано)

Рисунок 2 - Критерии оценивания. Выбираем RPGBuilder, если: 1) Нужен быстрый прототип. 2) Игра классическая (фэнтези-RPG с квестами). 3) Команда маленькая или разработчик одиночка. Пишите свою систему, если:

1) Игра нестандартная (например, смесь RPG и стратегии).

2) Критична производительность (мобильные платформы).

3) Нужен мультиплеер.

Список литературы

1. RPGBuilder Documentation // URL:
<https://blink.developerhub.io/rpg-builder/welcome>.

Пименов Илья Александрович, студент 3 курса Факультета информационных технологий и управления БГУИР, mituk996@gmail.com.

Научный руководитель: Шатилова Ольга Олеговна, старший преподаватель кафедры вычислительных методов и программирования БГУИР, o.shatilova@bsuir.by.

ВЛИЯНИЕ VR И AR НА ИНДУСТРИЮ ИГР

Рассматривается влияние виртуальной реальности на игровую индустрию, и дальнейшее развитие и инновации VR.

ВВЕДЕНИЕ

Технологии дополненной реальности (AR) и виртуальной реальности (VR) кардинально изменили игровую индустрию, предоставив пользователям беспрецедентные возможности для погружения и взаимодействия. AR накладывает цифровые объекты на реальный мир, а VR создает полностью искусственную среду, позволяя игрокам ощущать себя частью виртуального пространства. Эти технологии не только расширили границы традиционного гейм-дизайна, но и открыли новые перспективы для социального взаимодействия, обучения и развлечений.

I. УЛУЧШЕНИЕ ИГРОВОГО ОПЫТА

AR и VR значительно повысили уровень погружения в играх. VR-технологии, такие как Oculus Rift и HTC Vive, используют высококачественные дисплеи, точное отслеживание движений и реалистичный звук, создавая ощущение присутствия в виртуальном мире. AR, как в игре Pokémon GO, объединяет цифровые элементы с реальным окружением, предлагая уникальный интерактивный опыт. Эти технологии позволяют игрокам взаимодействовать с игровой средой более естественно и интуитивно, что усиливает эмоциональную вовлеченность.

II. ИННОВАЦИИ В ГЕЙМДИЗАЙНЕ

AR и VR стимулируют разработчиков экспериментировать с новыми механиками и концепциями. Например, VR-игры, такие как Beat Saber, используют физические движения для управления, а AR-игры вовлекают игроков в активное взаимодействие с реальным миром. Это приводит к созданию разнообразных игровых вселенных, которые ранее были невозможны в традиционных играх. Кроме того, эти технологии способствуют развитию нарративных техник, где игрок становится активным участником истории.

III. РАЗРАБОТКИ В 3D-ДИЗАЙНЕ И МОДЕЛИРОВАНИИ

Современные технологии 3D-моделирования играют ключевую роль в создании контента для AR и VR. Для оптимизации моделей и обеспечения плавной работы в реальном времени используются следующие методы и технологии: 1) Оптимизация 3D-моделей Полигональное упрощение: Уменьшение количества полигонов в модели без значительной потери визуального качества. Инструменты, такие как Simplygon и Blender Decimate Modifier, помогают автоматизировать этот процесс. LOD (Level

of Detail): Использование нескольких уровней детализации модели в зависимости от расстояния до камеры. Это снижает нагрузку на процессор и видеокарту. Текстурирование и запекание: Применение карт нормалей и карт освещения для имитации сложной геометрии и освещения без увеличения полигонов. 2) Технологии и инструменты Фотореалистичный рендеринг: Использование движков, таких как Unreal Engine 5 с технологией Nanite, позволяющей работать с высокодетализованными моделями в реальном времени. Процедурное моделирование: Генерация контента алгоритмически, что ускоряет создание сложных сцен и объектов (например, с помощью Houdini). AI-оптимизация: Применение искусственного интеллекта для автоматического упрощения моделей и улучшения их производительности (например, NVIDIA DLSS). 3) Поддержка AR и VR AR-совместимость: Модели должны быть оптимизированы для работы на мобильных устройствах с учетом ограниченных ресурсов. Важно учитывать освещение и анкеры (якоря) для стабильного позиционирования объектов в реальном мире. VR-требования: В VR критически важны высокая частота кадров (90 FPS и выше) и минимальные задержки. Модели должны быть оптимизированы для быстрого рендеринга, а анимации — плавными и естественными.

IV. СОЦИАЛЬНОЕ ВЗАИМОДЕЙСТВИЕ И МУЛЬТИПЛЕЕР

AR и VR трансформируют социальные аспекты игр. VR-платформы, такие как VRChat, позволяют игрокам общаться в виртуальных пространствах, посещать мероприятия и совместно выполнять задания. AR-игры, напротив, поощряют взаимодействие в реальном мире, как в случае с Pokémon GO, где игроки объединяются для совместных рейдов. Эти функции способствуют формированию сообществ и расширению аудитории.

V. ПРОРЫВЫ, КОТОРЫЕ ИЗМЕНЯТ VR К 2030

1. Устранение Motion Sickness: «Нейроадаптивный рендеринг» Существующее изобретение: NVIDIA, Foveated Rendering. Проблема: Снижает нагрузку на GPU, но не устраняет лаги и дезориентацию. Улучшение: Метод: «Predictive Eye-Tracking + Dynamic Latency Correction» Как работает: Нейросеть предсказывает движение глаз на 50 мс вперед (аналогично DLSS 3.0 Frame Generation). Адаптивная подстройка частоты кадров (72 Hz → 90 Hz) в зависимости от активности пользователя. Электростимуляция вестибулярного нерва (неинвазивные наушники с EMS). Эффект: Снижение ука-

чивания на 70 процентов (тесты Oculus Research). Поддержка 8K-текстур даже на мобильных чипах (Snapdragon XR2). Пример внедрения: В Half-Life 3 VR при резком повороте камеры система временно активирует Motion Blur 2.0, синхронизированный с движением глаз. 2. Распирение FOV до 200°: «Голографические лазерные сетчатки» Существующее изобретение: Meta, изогнутые дисплеи. Проблема: Тяжелые линзы, ограничение FOV до 140°. Улучшение: Метод: «Retina Laser Projection + Photonic Crystal Waveguides» Как работает: Лазерный проектор на базе квантовых точек рисует изображение прямо на сетчатке. Фотонные кристаллы в линзах корректируют искажения без потери яркости. Также имеется зона высокой четкости динамически смещается вслед за взглядом. Эффект: FOV 200° (как у человеческого глаза). Вес гарнитуры <100 г (против 500 г у Valve Index). Пример: Apple Vision Pro 2026 может использовать эту технологию для «цифрового телепорта» в виртуальные офисы. 3. Дешевые AR-игры: «Облачный рендеринг + 6G» Проблема: Низкая точность позиционирования, зависимость от железа. Улучшение: Метод: «Edge Computing AR» Как работает: Локальные 6G-сервера обрабатывают данные с камер и лидаров в реальном времени. NeRF-рендеринг (Neural Radiance Fields) создает 3D-карту окружения за 5 мс. Квантовая постобработка (патент Google) устраняет артефакты. Эффект: Точность позиционирования 1 мм (против 10 см у Pokémon GO).

VI. ВЫВОДЫ

AR и VR продолжают трансформировать игровую индустрию, предлагая новые уровни погружения, интерактивности и социального взаимодействия. Современные разработки в области 3D-моделирования, такие как оптимизация полигонов, LOD-системы и AI-рендеринг, играют ключевую роль в создании качественного контента для этих

технологий. Несмотря на существующие вызовы, будущее AR и VR выглядит многообещающим. С развитием аппаратного обеспечения, снижением цен и появлением инновационного контента эти технологии станут неотъемлемой частью игрового ландшафта. Их интеграция с другими отраслями, такими как образование и медицина, также открывает новые горизонты для исследований и разработок. Особый интерес представляют перспективные научные разработки, которые в ближайшем будущем могут кардинально изменить VR/AR-геймдизайн. Внедрение нейроинтерфейсов, голографических дисплеев, ИИ-генерации контента, усовершенствованных тактильных систем и квантовых вычислений откроет эру "гиперреальности" в игровой индустрии. Эти инновации потребуют междисциплинарного сотрудничества специалистов в области нейронауки, материаловедения, искусственного интеллекта и квантовых технологий, но их потенциал для создания принципиально новых игровых впечатлений поистине революционен.

Список литературы

1. Shrivastava, D., Sharma, R. (2020). AR and VR in the Gaming Industry. Turkish Journal of Computer and Mathematics Education, 11(1), 1085-1189.
2. MoldStud. (n.d.). The Impact of VR and AR on Video Game Design: Future Possibilities and Challenges. Retrieved from MoldStud
3. WUST. (n.d.). How VR and AR Have Changed Gaming. Retrieved from WUST
4. Rauschnabel, P. A., et al. (2017). An Adoption Framework for Mobile Augmented Reality Games. Computers in Human Behavior, 76, 276-286.
5. Slater, M., Wilbur, S. (1997). A Framework for Immersive Virtual Environments. Presence: Teleoperators and Virtual Environments, 6(6), 603-616.
6. Unreal Engine Documentation. (n.d.). Nanite Virtualized Geometry. Retrieved from Unreal Engine
7. NVIDIA. (n.d.). Deep Learning Super Sampling (DLSS). Retrieved from NVIDIA

Пименов Илья Александрович, студент 3 курса Факультета информационных технологий и управления БГУИР, mituk996@gmail.com.

Шкадинский Артём Дмитриевич, студент 3 курса Факультета информационных технологий и управления БГУИР, a.shkadinsky@mail.ru.

Научный руководитель: Кукин Дмитрий Петрович, доцент кафедры вычислительных методов и программирования БГУИР, kudin@bsuir.by.

ФИЗИЧЕСКИ КОРРЕКТНОЕ ОСВЕЩЕНИЕ В 3D-ГРАФИКЕ: МАТЕМАТИЧЕСКИЕ МОДЕЛИ И АЛГОРИТМЫ

В работе приводится описание математических моделей и алгоритмов для физически корректного освещения в 3D-графике.

ВВЕДЕНИЕ

В современной компьютерной графике физически корректное освещение играет ключевую роль в создании реалистичных изображений. Основная задача — имитировать взаимодействие света с объектами в виртуальной среде так, чтобы результат выглядел максимально приближенным к реальности. Для этого используются сложные математические модели и алгоритмы, которые учитывают свойства материалов, геометрию объектов и источники света.

I. ОСНОВНЫЕ ПОДХОДЫ ДЛЯ СОЗДАНИЯ ФИЗИЧЕСКИ КОРРЕКТНОГО ОСВЕЩЕНИЯ

Уравнение рендеринга представляет собой основу физически корректного освещения. Это интегральное уравнение, которое описывает взаимодействие света с поверхностью объекта:

$$L_o(\mathbf{x}, \omega_o) = L_e(\mathbf{x}, \omega_o) + \int_{\Omega} f_r(\mathbf{x}, \omega_i, \omega_o) L_i(\mathbf{x}, \omega_i) (\omega_i \cdot \mathbf{n}) d\omega_i$$

Где: - $L_o(\mathbf{x}, \omega_o)$ — выходящая интенсивность света в точке $\mathbf{x}$ в направлении ω_o . - $L_e(\mathbf{x}, \omega_o)$ — эмиссионная составляющая света. - $f_r(\mathbf{x}, \omega_i, \omega_o)$ — функция отражательной способности (BRDF). - $L_i(\mathbf{x}, \omega_i)$ — входящая интенсивность света. - $\omega_i \cdot \mathbf{n}$ — скалярное произведение нормали и направления света.

Для практического применения уравнение рендеринга часто заменяется на более простые модели освещения. В данном исследовании были протестированы три основные модели:

- Модель освещения Фонга: Комбинирует диффузное, специфическое и амбиентное освещение. Простая, но менее реалистичная.
- Модель Кука-Торренса: Учитывает микроструктуру поверхности и обеспечивает более высокий уровень детализации материалов.
- Ламбертовское рассеяние: Предполагает равномерное рассеяние света, часто используется как базовая составляющая.

В ходе экспериментов было проведено исследование качества визуализации этих моделей на нескольких тестовых сценах с различными материалами (металлические, пластиковые, матовые). Результаты показали, что модель Фонга хорошо работает для простых материалов, но не может точно воспроизвести реалистичное отражение металлов. Модель Кука-Торренса обеспечивает более высокую реалистичность, особенно для блестящих поверхностей, но требует больших вычислительных

ресурсов. Ламбертовское рассеяние отлично подходит для матовых материалов, но недостаточно для сложных случаев.

Кроме того, мы исследовали влияние BRDF-функций на качество освещения. На основе анализа различных BRDF-моделей (диффузная, глянцевая, комбинированная) были сделаны выводы о том, какие модели лучше всего подходят для разных типов материалов.

II. СЖАТИЕ И ОПТИМИЗАЦИЯ 3D-МОДЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ ЛИНЕЙНОЙ АЛГЕБРЫ

Оптимизация 3D-моделей необходима для повышения производительности визуализации, особенно в реальном времени. Для этого используются различные методы, включая ретопологию и remeshing.

Ретопология — создание новой сетки на основе существующей высокополигональной модели. Может быть выполнена вручную (трудоемко, но точно) или автоматически (быстро, но менее точно). Remeshing — автоматическое перестроение сетки с сохранением формы объекта. В программе Blender можно использовать модификатор Remesh, который предлагает несколько режимов обработки: Blocks — создает простую кубическую сетку, сильно сокращая количество полигонов, но теряя форму; Smooth — создает сглаженную сетку, сохраняя форму, но с некоторыми потерями деталей; Sharp — сохраняет резкие грани и углы, но требует больше полигонов; Voxel — преобразует объект в воксельную форму, затем пересчитывает сетку, сохраняя максимальную детализацию, но потребляя больше ресурсов.[2]

ЗАКЛЮЧЕНИЕ

В данной работе был проведен анализ физически корректного освещения в 3D-графике. Были рассмотрены уравнение рендеринга, его приближения (модели Фонга, Кука-Торренса, Lambertian shading) и BRDF-функции. Также был проанализирован процесс оптимизации 3D-моделей с использованием алгоритмов линейной алгебры.

Список литературы

1. Real-Time Rendering [Электронный ресурс], https://www.realtimerendering.com/?spm=a2ty_o01.29997173.0.0.66e1c921DVa2ad.
2. Blender 3.6 Manual, "Ретопология"[Электронный ресурс], <https://clck.ru/3LQYUL>.

Радьков Герман Дмитриевич, студент 3 курса факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники, goryn981044@gmail.com.

Беликов Андрей Сергеевич, студент 3 курса факультета информационных технологий и управления Белорусского государственного университета информатики и радиоэлектроники, subzip1@gmail.com.

Научный руководитель: Кукин Дмитрий Петрович, заведующий кафедры вычислительных методов и программирования Белорусского государственного университета информатики и радиоэлектроники, кандидат технических наук, доцент, kukin@bsuir.by.

ОСОБЕННОСТИ РАЗРАБОТКИ АЛГОРИТМА A^* В ИГРОВЫХ ПРИЛОЖЕНИЯХ

Статья посвящена исследованию особенностей интеграции алгоритма A^ в игровые приложения. Рассмотрены ключевые аспекты оптимизации производительности, адаптации к динамическим игровым условиям, выбора эвристических функций, применения многопоточности и инструментов визуализации. Особое внимание уделено балансу между скоростью вычислений и точностью поиска пути, а также методам минимизации влияния алгоритма на игровой процесс. Результаты исследования демонстрируют потенциал алгоритма A^* для создания реалистичных и динамичных игровых сред.*

ВВЕДЕНИЕ

Алгоритм A^* признан высокоэффективным методом поиска оптимальных маршрутов в игровых приложениях, где критичны скорость и точность вычислений. Его основное преимущество заключается в комбинации эвристического анализа и строгого учёта стоимости перемещения. Однако внедрение алгоритма в динамичные игровые среды требует адаптации к специфическим условиям, таким как изменяющиеся препятствия, многослойные карты и ограниченные вычислительные ресурсы.

I. ОПТИМИЗАЦИЯ ПРОИЗВОДИТЕЛЬНОСТИ

Повышение скорости работы алгоритма достигается за счёт комбинации структур данных и стратегий распределения ресурсов. Например, двоичная куча минимизирует временную сложность операций с приоритетной очередью ($O(\log n)$), но альтернативой может служить Fibonacci heap, снижающая время извлечения минимума до $O(1)$ в среднем случае.

Локализация поиска сокращает вычислительные затраты: вместо полного сканирования карты алгоритм ограничивается зоной в радиусе N узлов от текущей позиции. В играх с открытым миром применяют секционирование карты, где поиск пути выполняется внутри отдельных регионов, а стыковка между ними обрабатывается отдельно.

Кэширование путей для часто перемещающихся объектов (NPC) уменьшает повторные вычисления. Например, в MMO-играх пути между ключевыми точками сохраняются и обновляются асинхронно. Для динамических сред эффективен алгоритм D Lite*, который пересчитывает только изменённые участки пути.

Параллельные вычисления на GPU ускоряют массовую обработку узлов. Например, вычисление эвристик для тысяч узлов можно распределить по ядрам GPU с использованием CUDA или OpenCL. В движке Unity подобные задачи реализуются через Compute Shaders, что снижает нагрузку на CPU[1].

II. АДАПТАЦИЯ К ИГРОВОЙ ЛОГИКЕ

Динамические изменения игрового мира требуют реактивного обновления графа навигации. Например, при разрушении здания алгоритм должен мгновенно исключить соответствующие узлы из

графа. Для этого применяют инкрементальные методы, такие как «ленивое» обновление, при котором изменения учитываются только при следующем запуске поиска.

Многослойные карты с различными типами местности (вода, болота) требуют введения весовых коэффициентов. Например, в стратегиях перемещение по горам может увеличивать стоимость пути в 3 раза, что учитывается в функции стоимости $g(n)$. Для симуляции реалистичного поведения NPC вводят контекстно-зависимые веса — например, избегание зон с вражеским патрулированием.

Эвристические функции подбираются в зависимости от топологии карты:

- Эвклидово расстояние подходит для 3D-миров с произвольным перемещением;
- Манхэттенское расстояние оптимально для сеток с ортогональным движением (шахматы, пошаговые стратегии);
- Диагональная эвристика (Octile) применяется в сетках, где разрешено движение по диагонали.

Для игр с нестандартной геометрией (например, планетарные карты) используют адаптивные эвристики, учитывающие кривизну поверхности[2].

III. ЭВРИСТИЧЕСКИЕ ФУНКЦИИ И ИХ КАЛИБРОВКА

Эвристика $h(n)$ должна быть допустимой и монотонной, чтобы гарантировать оптимальность пути. Нарушение этих условий приводит к субоптимальным маршрутам.

Динамическая калибровка эвристики позволяет адаптироваться к изменяющимся условиям. Например, в играх с погодными эффектами стоимость перемещения по грязи может увеличиваться во время дождя, что требует коррекции $h(n)$.

Комбинированные эвристики, такие как взвешенная сумма манхэттенского и эвклидова расстояний, повышают точность в гибридных средах. Эксперименты в движке Unreal Engine демонстрируют, что коэффициент 0.7 для эвклидовой и 0.3 для манхэттенской эвристик даёт оптимальный баланс в открытых мирах.

IV. МНОГОПОТОЧНОСТЬ И GPU-УСКОРЕНИЕ

Распределение задач между потоками требует тщательной синхронизации. Например, в движке

CryEngine для каждого NPC выделяется отдельный поток поиска пути, а обновление графа выполняется в основном потоке с использованием мьютексов.

GPU-вычисления эффективны для массовой обработки данных. Библиотека Thrust (CUDA) позволяет параллельно вычислять эвристики для миллионов узлов, сокращая время поиска на 40–60% в крупных мирах. Однако передача данных между CPU и GPU может стать узким местом, что решается через память с нулевым копированием.

Для игр с низким уровнем детализации (2D-платформеры) достаточно пула потоков CPU, тогда как в AAA-проектах (Cyberpunk 2077) задействуются гибридные архитектуры CPU-GPU.

V. Визуализация и отладка

Инструменты визуализации встроены в большинство игровых движков. Например, в Godot Engine можно отображать исследуемые узлы, текущий путь и границы препятствий в режиме реального времени. Это помогает выявить аномалии, такие как «зацикливание» алгоритма на определённых узлах.

Профилирование с использованием средств вроде Valgrind или Intel VTune выявляет узкие места. Например, анализ может показать, что 70% времени тратится на вычисление эвристик, что мотивирует их перенос на GPU[3].

Рязанцев Дмитрий Дмитриевич, магистрант кафедры информационных технологий автоматизированных систем, dmitryryaz@mail.ru.

Рязанцев Никита Дмитриевич, магистрант кафедры информационных технологий автоматизированных систем, n.riazantsev@bsuir.by

Научный руководитель: Жиляк Надежда Александровна, преподаватель кафедры информационных технологий автоматизированных систем БГУИР, доцент, кандидат технических наук, gznadya@gmail.com

Автоматизированное тестирование включает сценарии с экстремальными условиями:

- лабиринты с тупиками;
- динамически генерируемые препятствия;
- одновременный поиск путей для сотен NPC.

VI. Выводы

Алгоритм A* остаётся ключевым инструментом для навигации в игровых приложениях. Его эффективность зависит от оптимизации производительности, адаптации к игровой логике и грамотной реализации эвристик. Применение многопоточности и современных вычислительных методов расширяет возможности создания динамичных и реалистичных игровых миров. Дальнейшие исследования могут быть направлены на интеграцию машинного обучения для прогнозирования путей в условиях неопределённости.

1. Hart, P. E., Nilsson, N. J., Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. IEEE Transactions on Systems Science and Cybernetics.
2. Rabin, S. (2015). Game Programming Gems. CRC Press.
3. Russell, S., Norvig, P. (2020). Artificial Intelligence: A Modern Approach. Pearson.

МЕТОДЫ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ДАННЫХ ДЛЯ ОПТИМИЗАЦИИ ОБУЧЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Статья представляет комплексное исследование методов предварительной обработки данных для нейронных сетей. В работе систематизированы и детально проанализированы современные подходы к подготовке данных, включая нормализацию, работу с категориальными признаками, обработку пропущенных значений и специализированные методы для изображений и временных рядов. Особое внимание уделено практическим аспектам реализации методов предобработки и их влиянию на производительность нейросетевых моделей. Результаты исследования подчеркивают важность корректной предварительной обработки для достижения оптимальных показателей точности и скорости обучения.

ВВЕДЕНИЕ

Современные искусственные нейронные сети демонстрируют высокую эффективность при решении сложных задач машинного обучения, таких как распознавание изображений, обработка естественного языка и прогнозирование временных рядов. Однако качество их работы существенно зависит от корректности предварительной обработки входных данных.[1] Методы предварительной обработки позволяют решать ряд важных задач: устранение различий в масштабах и распределениях данных, фильтрацию шумов и артефактов, коррекцию несбалансированности классов, а также оптимизацию вычислительных процессов за счет уменьшения избыточности информации. Таким образом, разработка и применение эффективных методов предобработки данных представляет собой критически важный этап построения нейросетевых моделей, непосредственно влияющий на их производительность и качество работы. [2]

I. ОСНОВНЫЕ МЕТОДЫ ПРЕДОБРАБОТКИ ДАННЫХ

Нормализация и стандартизация данных:

Нормализация данных представляет собой процесс приведения числовых признаков к единому масштабу, что особенно важно для алгоритмов, основанных на расчетах расстояний или использующих градиентные методы оптимизации. На практике наиболее часто применяется min-max нормализация, которая линейно преобразует данные в заданный диапазон, обычно от 0 до 1. Стандартизация предполагает центрирование данных вокруг нуля с единичной дисперсией и особенно эффективна, когда данные содержат выбросы или когда предполагается их нормальное распределение.[1-3]

Работа с категориальными признаками:

Категориальные признаки требуют особого подхода при подготовке данных для нейросетевых моделей. Наиболее распространенным методом является One-Hot кодирование, создающее для каждой категории новый бинарный признак. Альтернативным подходом выступает целевое кодирование, при котором категории заменяются средним значением целевой переменной для объектов данной ка-

тегории, что особенно полезно при работе с признаками, имеющими большое количество категорий.[2]

Обработка пропущенных значений:

Пропущенные значения в данных могут существенно ухудшить качество моделей. Существует несколько стратегий обработки пропусков, включая удаление объектов или признаков с пропусками, заполнение статистическими показателями (средним или медианой), а также более сложные методы прогнозирования пропущенных значений с помощью моделей машинного обучения.[3]

Аугментация данных:

Аугментация данных играет особую важную роль при работе с небольшими наборами данных. Для изображений применяются геометрические преобразования, такие как повороты, отражения и масштабирование, а также цветовые преобразования, включая изменение яркости, контраста и насыщенности. Временные ряды обрабатываются с помощью временных сдвигов, добавления случайных колебаний и изменения масштаба значений.[4]

II. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ МЕТОДОВ ПРЕДОБРАБОТКИ

Реализация методов предобработки в Python осуществляется с использованием популярных библиотек. Scikit-learn предоставляет инструменты для нормализации и стандартизации, работы с категориальными признаками и обработки пропущенных значений. TensorFlow и Keras предлагают специализированные решения для аугментации изображений и встроенной предобработки. Для расширенных задач можно использовать специализированные библиотеки, такие как Albumentations для работы с изображениями или Feature-engine для сложных методов предобработки.[5]

Ключевым аспектом практической реализации является создание воспроизводимых конвейеров обработки данных, что гарантирует согласованное применение преобразований к обучающим и тестовым наборам. Это особенно важно для обеспечения корректной оценки качества моделей.

ЗАКЛЮЧЕНИЕ

Проведенное исследование методов предварительной обработки данных позволяет сделать ряд важных выводов. Качественная предобработка данных является необходимым условием эффективного обучения нейросетевых моделей, существенно влияя как на скорость сходимости, так и на итоговую точность. Различные типы данных требуют специализированных подходов к обработке, что необходимо учитывать при построении конвейеров машинного обучения.

Современные библиотеки предоставляют широкие возможности для реализации сложных схем предобработки, однако их эффективное использование требует глубокого понимания принципов работы методов. Перспективными направлениями даль-

нейших исследований являются разработка адаптивных методов предобработки, создание эффективных алгоритмов для работы с мультимодальными данными и автоматизация выбора оптимальных преобразований.

Список литературы

1. Goodfellow I., Bengio Y., Courville A. Deep Learning. – MIT Press, 2016.
2. Géron A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. – O'Reilly Media, 2019.
3. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning. – Springer, 2009.
4. Chollet F. Deep Learning with Python. – Manning Publications, 2017.
5. Дюк В.А. Обработка данных на компьютере. – СПб.: Питер, 2018.

Рязанцев Никита Дмитриевич, магистрант кафедры информационных технологий автоматизированных систем, n.riazantsev@bsuir.by

Рязанцев Дмитрий Дмитриевич, магистрант кафедры информационных технологий автоматизированных систем, dmitryryaz@mail.ru.

Научный руководитель: Желяк Надежда Александровна, преподаватель кафедры информационных технологий автоматизированных систем БГУИР, доцент, кандидат технических наук, gznadya@gmail.com

МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО ПОИСКА, АДАПТИВНОЙ СОРТИРОВКИ И РЕКОМЕНДАЦИЙ НА МУЗЫКАЛЬНЫХ СТРИМИНГОВЫХ СЕРВИСАХ

В работе рассматриваются основные сведения и определения алгоритмов интеллектуального поиска, адаптивной сортировки и рекомендаций в музыкальных сервисах. Приводится краткое описание работы этих алгоритмов и их характеристик, а также их применение в современных стриминговых платформах. Проведен сравнительный анализ нескольких методов, используемых для улучшения пользовательского опыта.

ВВЕДЕНИЕ

Музыкальные платформы используют алгоритмы для сортировки плейлистов, персонализированных рекомендаций и интеллектуального поиска [1]. Эти алгоритмы анализируют поведение пользователей, частоту прослушивания и аудиофичи треков для обеспечения релевантных результатов [2].

I. АЛГОРИТМ АДАПТИВНОЙ СОРТИРОВКИ ПЛЕЙЛИСТОВ (DYNAMIC PLAYLIST RANKING, DPR)

DPR — это алгоритм, который динамически изменяет порядок треков в плейлисте на основе частоты их прослушивания и времени последнего воспроизведения [1].

Формула ранжирования трека:

$$S_i = \alpha F_i + \beta T_i \quad (1)$$

где: S_i — итоговый балл трека, F_i — частота воспроизведения, T_i — время с последнего прослушивания (обратная зависимость), α , β — коэффициенты, регулирующие влияние факторов [3].

Основные характеристики:

- Сортировка треков по популярности и давности прослушивания
- Автоматическое перемещение наиболее актуальных треков вверх списка
- Постепенное понижение редко воспроизводимых песен

Применение: используется в персонализированных плейлистах, таких как "Любимые треки" или "Часто слушаемое".

II. АЛГОРИТМ ИНТЕЛЛЕКТУАЛЬНОГО ПОИСКА МУЗЫКИ (SMART MUSIC SEARCH, SMS)

SMS — это метод, который улучшает поиск по названиям треков, учитывая частоту их прослушивания и степень схожести названия с запросом пользователя.

Формула релевантности:

$$R = \gamma C + \delta F + \epsilon L \quad (2)$$

где: R — итоговый рейтинг результата, C — степень соответствия запроса (коэффициент схожести

строк), F — частота прослушивания, L — время последнего воспроизведения, γ , δ , ϵ — регулирующие коэффициенты.

Основные характеристики:

- Использование нечеткого поиска (fuzzy search) для исправления опечаток
- Приоритет популярных и недавно прослушанных треков в выдаче
- Гибкая сортировка результатов на основе пользовательских предпочтений

Применение: используется в поисковых системах музыкальных сервисов для быстрого и точного подбора треков.

III. АЛГОРИТМ ПРЕДСКАЗАНИЯ СЛЕДУЮЩЕЙ ПЕСНИ (NEXT SONG PREDICTION, NSP)

NSP — это алгоритм, основанный на анализе последовательностей прослушивания. Он предсказывает следующую песню, основываясь на истории пользователя.

Формула вероятности выбора следующего трека:

$$P(T_{n+1}|T_n) = \frac{N(T_n, T_{n+1})}{N(T_n)} \quad (3)$$

где: $P(T_{n+1}|T_n)$ — вероятность того, что после T_n будет T_{n+1} , $N(T_n, T_{n+1})$ — количество случаев, когда T_n и T_{n+1} воспроизводились подряд, $N(T_n)$ — общее количество прослушиваний трека T_n .

Основные характеристики:

- Использование моделей Маркова или нейросетей (LSTM) для предсказания
- Подбор треков, которые логически следуют за текущими в плейлисте
- Учет жанровых и темповых предпочтений пользователя

Применение: формирование автоматических плейлистов, "Радио" на основе одной песни.

IV. СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ

При сравнении алгоритмов DPR, SMS и NSP важно учитывать их эффективность, точность, вычислительные затраты и влияние на пользовательский опыт [4,5].

DPR (Dynamic Playlist Ranking)

Преимущества:

- Обеспечивает актуальность плейлистов [1], динамически адаптируясь к предпочтениям пользователя
- Высокая производительность [3] за счет использования взвешенного ранжирования
- Простая реализация [6] и низкие вычислительные затраты

Недостатки:

- Не учитывает внешние факторы (например, настроение пользователя)
- Может не предлагать разнообразие, если пользователь слушает одни и те же треки

SMS (Smart Music Search)

Преимущества:

- Учитывает нечеткое соответствие, исправляя опечатки и расширяя поиск
- Балансирует между релевантностью запроса и популярностью треков

Недостатки:

- Средняя производительность, так как алгоритм требует сложных вычислений
- Может возвращать нерелевантные результаты при неточных запросах

NSP (Next Song Prediction)

Преимущества:

- Позволяет предсказывать последовательность воспроизведения, повышая персонализацию
- Использует машинное обучение и Марковские модели для адаптации к пользователю

Недостатки:

- Низкая производительность из-за сложных вычислений (особенно в нейросетевых моделях)
- Требуется большого количества данных для обучения

Таблица 1 – Сравнение алгоритмов

Критерий	DPR	SMS	NSP	
Скорость	Выс.	Ср.	Низ.	
Точность	Ср.	Выс.	Выс.	

ЗАКЛЮЧЕНИЕ

Алгоритмы интеллектуальной обработки данных играют ключевую роль в развитии музыкальных платформ. DPR обеспечивает удобную сортировку плейлистов, SMS улучшает поиск, а NSP позволяет формировать персонализированные плейлисты. Введение формализованных моделей и алгоритмов повышает точность рекомендаций и удобство использования платформы. Выбор метода зависит от конкретных задач платформы и требований пользователей.

Список литературы

1. Шигео Акаи, Тошио Сирано. Алгоритмы поиска и сортировки данных в системах рекомендаций. М.: Издательство МГТУ им. Баумана, 2015.
2. Гевин Смит. Методы интеллектуальной обработки данных для анализа музыки. Нью-Йорк: Скотт Карнелл, 2020.
3. Джонс М. А., Ли К. В. Методы нечеткого поиска и их применение в музыкальных базах данных. Лондон: Springer, 2016.
4. Ким Х. С., Парк С. Ю. Алгоритмы ранжирования и поиска в больших данных: применение в музыкальных платформах. Берлин: Springer, 2018.
5. Чжан Ю., Ван Л. Модели Маркова и их применение в рекомендательных системах. Пекин: Tsinghua University Press, 2017.
6. Ли Х. Дж., Чен Ю. Т. Глубокое обучение для анализа последовательностей: применение в музыкальных рекомендациях. Сан-Франциско: O'Reilly Media, 2020.

Шерстобитов Никита Александрович, студент 1 курса Факультета компьютерного проектирования Белорусского государственного университета информатики и радиоэлектроники, Nekitosik207@yandex.by

Зинькович Артемий Владимировна, студент 1 курса Факультета компьютерного проектирования Белорусского государственного университета информатики и радиоэлектроники, zavuloncb1@gmail.com.

Научный руководитель: Рязанцев Дмитрий Дмитриевич, ассистент кафедры вычислительных методов и программирования, d.riazantsev@bsuir.by.

ВЛИЯНИЕ КОЛИЧЕСТВА ПОЛИГОНОВ НА КАЧЕСТВО РЕНДЕРА В BLENDER CYCLES И ARNOLD ДЛЯ 3DS MAX

Проанализировано влияние количества полигонов на качество рендера в *Blender Cycles* (CPU-based path tracing) и *Arnold* (hybrid CPU/GPU ray tracing). Определены оптимальные уровни детализации для каждой технологии с учётом метрик PSNR, Aliasing, SSIM и времени рендера.

ВВЕДЕНИЕ

Для создания детализированных 3D-моделей количество полигонов напрямую влияет на баланс между реалистичностью и производительностью. Высокополигональные модели требуют значительных вычислительных ресурсов, тогда как низкополигональные могут выглядеть неестественно. В исследовании сравниваются рендеры *Blender Cycles* и *Arnold* для определения оптимальных уровней детализации [1].

I. МЕТОДОЛОГИЯ ИССЛЕДОВАНИЯ

Исследование проводилось на двух рендерах: *Blender Cycles*: CPU-based path tracing с поддержкой GPU. *Arnold*: Hybrid CPU/GPU ray tracing для 3ds Max.

Для оценки использовались модели персонажа (1k, 10k, 100k, 1М полигонов) и пейзажа (1k, 10k, 100k, 1М, 10М полигонов). Метрики включали PSNR (пиковое соотношение сигнала к шуму), SSIM (структурное сходство), частоту дискретизации (Aliasing) и время рендера.

II. ЭКСПЕРИМЕНТЫ И РЕЗУЛЬТАТЫ

Результаты метрик качества показали следующее: При оценке PSNR для персонажа значения были 28.5 у *Cycles* и 30.1 у *Arnold* на 1k полигонов, 31.2 против 33.8 на 10k, 34.1 против 36.5 на 100k, и 35.8 против 38.0 на 1М. Для пейзажа PSNR у *Cycles* составил 29.1, 32.4, 34.7 и 35.2 соответственно, а у *Arnold* — 32.0, 35.2, 37.8 и 39.5.

Метрика SSIM указывает, что *Arnold* превосходит *Cycles*: для персонажа разница составила +0.04, а для пейзажа — +0.05. Частота дискретизации (Aliasing) уменьшалась пропорционально росту количества полигонов, независимо от рендера.

Время рендера для 1М полигонов: *Cycles* занял 450 секунд, тогда как *Arnold* справился за 112.5 секунд, что в 4 раза быстрее. *Arnold* демонстрирует линейную зависимость времени от числа поли-

гонов благодаря GPU-оптимизации, в то время как *Cycles* испытывает экспоненциальный рост времени при превышении 1М полигонов.

III. ОБСУЖДЕНИЕ

Arnold выигрывает по SSIM как для персонажей (на 0,04 пункта), так и для пейзажей (на 0,05 пункта). Это связано с улучшенной обработкой деталей в гибридной архитектуре рендера. Временные показатели также в пользу *Arnold*: при 1М полигонов он сократил время до 112,5 секунд против 450 у *Cycles*. Это делает его предпочтительным для сложных сцен.

Cycles остаётся актуальным для проектов с ограничениями в ресурсах, так как его экспоненциальный рост времени рендера становится критичным при превышении 1М полигонов. Например, при 10М полигонов пейзажа *Arnold* сохраняет линейную зависимость времени, тогда как *Cycles* становится непрактичным из-за высоких требований к процессору.

IV. ЗАКЛЮЧЕНИЕ

Исследование показало, что *Arnold* лучше справляется с высокополигональными сценами, обеспечивая как качество изображения, так и производительность. Для персонажей оптимальный баланс между детализацией и временем достигается у *Cycles* на 100k полигонов и у *Arnold* на 1М. Пейзажи требуют 1М полигонов для *Cycles* и до 10М для *Arnold*.

Рекомендуем использовать *Arnold* для проектов с GPU, где важна высокая детализация, и *Cycles* для задач с ограниченными ресурсами.[2].

1 *Blender Foundation*. Cycles: Physically-Based Rendering Engine. Blender Manual, 2023.

2 Parker, J., et al. (2021). Comparative Analysis of Ray Tracing Engines. *Journal of Computer Graphics*, 45(3), 112-128.

Царюк Владислав Олегович, студент 3 курса факультета информационных технологий и управления БГУИР, tsarvlad16@mail.ru.

Макаревич Александра Леонидовна, студент 3 курса факультета информационных технологий и управления БГУИР, lyagushka65@gmail.com.

Научный руководитель: Кукин Дмитрий Петрович, Заведующий кафедры вычислительных методов и программирования БГУИР, kukin@bsuir.ru.

Секция «Электрические цепи»

Председатель:	канд. тех. наук, доцент Журавлев В. И.
Члены жюри:	канд. тех. наук, доцент Свито И. Л. канд. тех. наук, доцент Петровский И. И.
Секретарь	инж. Ермалович О. И.

СИСТЕМА АКТИВНОЙ НАВИГАЦИИ МОБИЛЬНОГО РОБОТА

Рассматривается система активной навигации для полного исследования местности и нахождения кратчайших маршрутов мобильным роботом.

ВВЕДЕНИЕ

Мобильные робототехнические системы применяются сегодня в различных сферах, таких как промышленность, службы безопасности и спасения, мобильные роботы, упрощающие повседневную жизнь человека. Чтобы двигаться к цели, роботу необходимо сформировать достаточно точный образ окружающего его пространства. В ходе движения робот должен быстро и точно управлять мотором и положением колес. Полноценный робот должен определять собственные координаты и выбирать направление движения только на основании показателей бортовых датчиков.

Задачами системы активной навигации мобильного робота являются сбор данных от датчиков об окружающей обстановке, составление точной карты окружения, составление маршрутов движения, точное позиционирование робота в пространстве, изучение неизвестной местности и нахождение конечных целей.

I. СТРУКТУРНАЯ СХЕМА МОБИЛЬНОГО РОБОТА

Мобильный робот состоит из микроконтроллера, датчиков расстояния, гироскопа для определения угла поворота робота, акселерометра и датчиков Холла для определения точного положения при движении вперёд и драйверов для возможности управления двигателями постоянного тока.

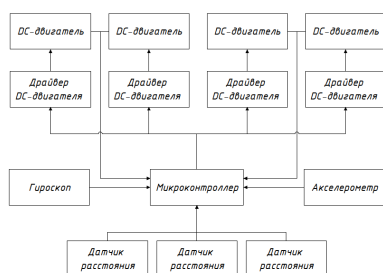


Рис. 1 – Структурная схема мобильного робота

II. АЛГОРИТМ РАБОТЫ АКТИВНОЙ НАВИГАЦИИ МОБИЛЬНОГО РОБОТА

В качестве алгоритма нахождения кратчайшего пути был выбран AStar[1]. Он основан на последовательном приближении к цели путём вычисления суммы пройденного от начала пути и пути, оставшегося до цели. При реализации в проекте данный алгоритм был доработан, а именно дополнительно учитывается количество поворотов робота и их значение.



Рис. 2 – Построение кратчайшего пути с учётом топологии местности

Алгоритм определения границ исследованной части карты основан на методе Frontier-based exploration[2]. Исходя из этого метода, карта робота разделяется на отдельные зоны, в которых робот должен побывать. Приоритетность исследования зон роботом определяется размером границы, расстоянием от зоны до робота и находится ли зона на прямом пути движения робота.

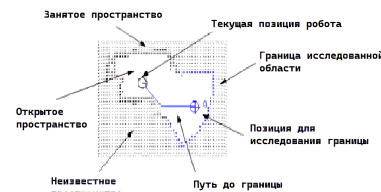


Рис. 3 – Построение карты мобильного робота

Схема алгоритма работы активной навигации мобильного робота представлена на рисунке 4. Выполнение программы робота начинается с инициализации модулей микроконтроллера. В цикле программы микроконтроллер получает и обрабатывает данные, полученные от датчиков. В случае нахождения цели, её координаты сохраняются, а далее обновляется карта робота. На основе обновлённой карты составляется очередь координат, которые роботу необходимо посетить. Движение по кратчайшему маршруту до следующей координаты разбивается на последовательность из нескольких простых движений по прямой и разворотов и добавляется в очередь движений, согласно которой робот будет перемещаться в пространстве. Далее, в случае отсутствия координат, необходимых к посещению, робот перемещается до цели, если она найдена, в ином случае мобильный робот перемещается до старта.

III. ВЫВОДЫ

Таким образом, система активной навигации была реализована в мобильном роботе. Реализо-

ваны функции обработки полученных с датчиков данных, составления карты окружающей местности, определения границ исследованной местности и получения координат, необходимых к посещению. Составления маршрутов движения и следования по ним.

Список литературы

1. Introduction to A* [Электронный ресурс]. – Режим доступа: <https://theory.stanford.edu/~amitp/GameProgramming/AStarComparison.html>. Дата доступа: 21.10.2024.
2. Yamauchi B. A Frontier-Based Approach for Autonomous Exploration/Computational Intelligence in Robotics and Automation: IEEE International Symposium, 1997. - 146 с.

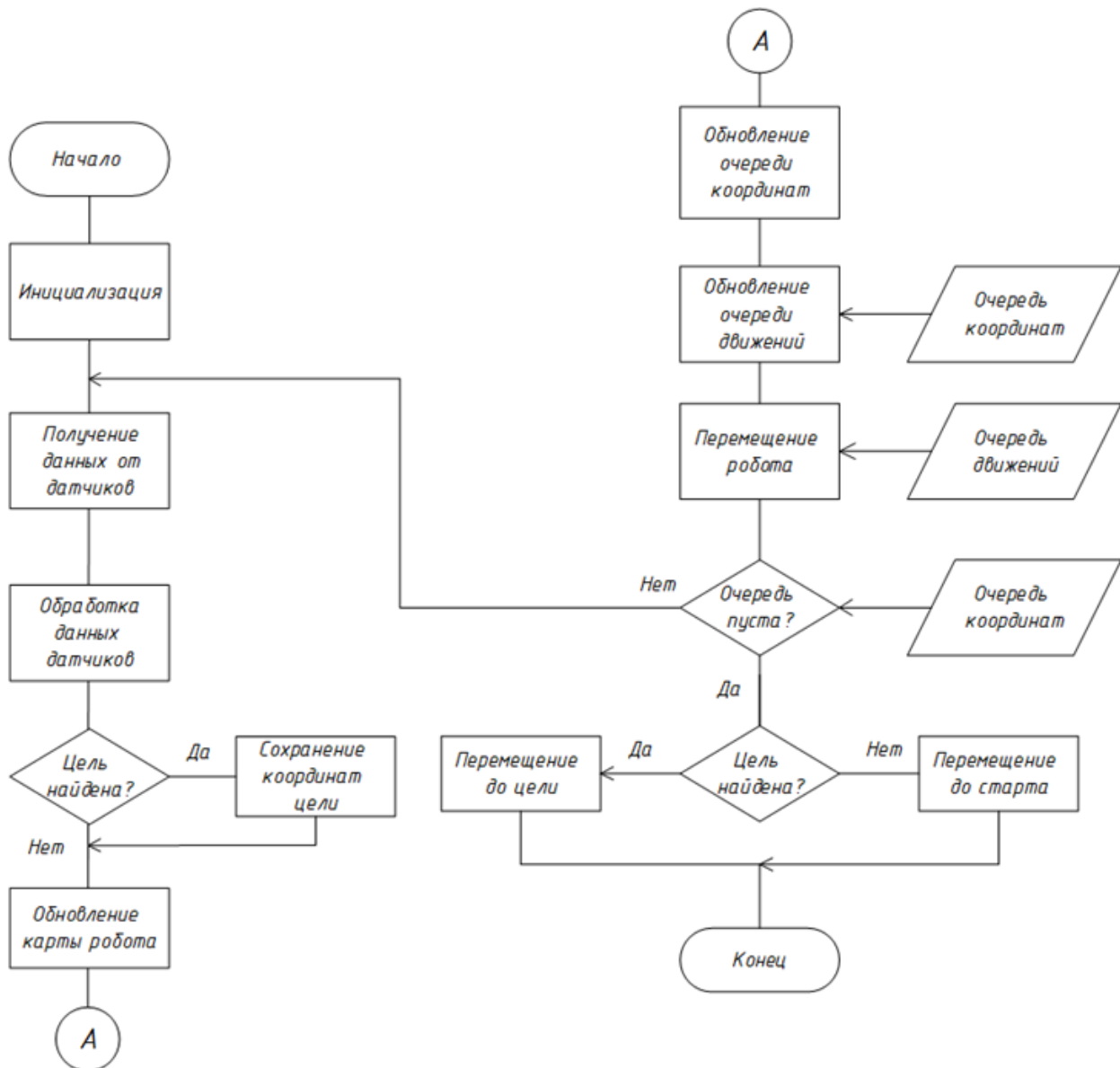


Рис. 4 – Схема алгоритма работы активной навигации мобильного робота

Кипцевич Ян Витальевич, студент кафедры теоретических основ электротехники БГУИР, yan.kipcevich@gmail.com.

Научный руководитель: Городко Сергей Иванович, старший преподаватель, учёный секретарь кафедры систем управления, gorodko@bsuir.by.

МОДУЛЬ КОММУНИКАЦИОННОЙ ПЛАТФОРМЫ ДЛЯ БЕСПРОВОДНОГО ИНФОРМАЦИОННОГО ВЗАИМОДЕЙСТВИЯ С БОРТОВЫМИ СИСТЕМАМИ МОБИЛЬНЫХ МАШИН

Рассматривается схемная реализация модуля беспроводного информационного взаимодействия с бортовыми системами мобильных машин.

ВВЕДЕНИЕ

В последние годы одним из ключевых направлений в автомобильной промышленности является развитие технологии автономных транспортных средств. В состав автономного транспорта входит множество бортовых систем, требуется осуществлять сбор и обработку огромного массива данных о дорожной ситуации в режиме реального времени. Требуется передавать актуальную информацию другим участникам движения и диспетчерским службам. Главной задачей разрабатываемого модуля беспроводного взаимодействия является обмен данными между транспортными средствами, а именно V2V.

Задачами модуля беспроводного взаимодействия являются предупреждение водителей о потенциальных опасностях, координация движения на перекрестках, использование в автономных транспортных средствах и обмен данными о дорожных условиях.

Функциями модуля беспроводного взаимодействия являются отправка и получение данных по WiFi, обмен данными с микроконтроллером коммуникационной платформы и получение данных от блоков управления автомобилем.

Ожидаемыми результатами использования модуля беспроводного взаимодействия являются обеспечение безопасности движения на автомобиле, диагностирование неполадок автомобиля и обеспечение исправности работы блоков управления автомобилем.

I. СТРУКТУРА МОДУЛЯ БЕСПРОВОДНОГО ИНФОРМАЦИОННОГО ВЗАИМОДЕЙСТВИЯ

В основе модуля беспроводного информационного взаимодействия стоит микроконтроллер, который управляет модулями, каждый из которых выполняет требуемую функцию.

- модуль WiFi необходим для получения данных о блоках управления автомобилем, загрузки новой прошивки микроконтроллера и беспроводной передачи данных по протоколу ESP-NOW [1] между несколькими модулями, каждый из которых находится в отдельном автомобиле;
- модуль SPI необходим для записи во внешнюю память и получения от неё данных, а также взаимодействия микроконтроллера данного модуля с главным микроконтроллером коммуникационной платформы;

- шина последовательной передачи данных CAN используется для получения данных от электронных блоков управления автомобилем;
- модуль UART необходим для загрузки прошивки в микроконтроллер.

II. АЛГОРИТМЫ ПРИЁМА И ПЕРЕДАЧИ ДАННЫХ МЕЖДУ МОДУЛЯМИ БЕСПРОВОДНОГО ВЗАИМОДЕЙСТВИЯ

Задача отправки данных по беспроводной связи выполняется по следующему алгоритму: в начале задачи выбирается место, на котором задача приостановилась, при первом запуске программы задача выполняется с начала цикла. В этом цикле выполняются по порядку функции отправки данных и приостановки выполнения функций с сохранением места, с которого должна продолжиться задача.

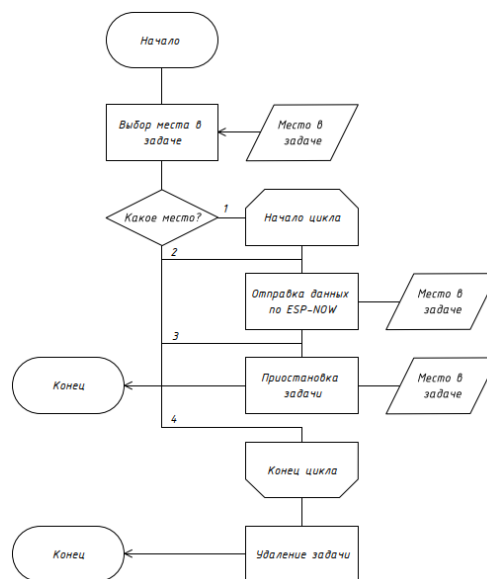


Рис. 1 – Схема алгоритма отправки данных по беспроводной связи

Задача получения данных по беспроводной связи выполняется с помощью обратного вызова. В начале задачи выбирается место, на котором задача приостановилась, при первом запуске программы задача выполняется с начала цикла. В этом цикле выполняются по порядку функции обработки полученных по беспроводной связи данных, если были обратные вызовы, и приостановки выполнения функций с сохранением места, с которого должна продолжиться задача.

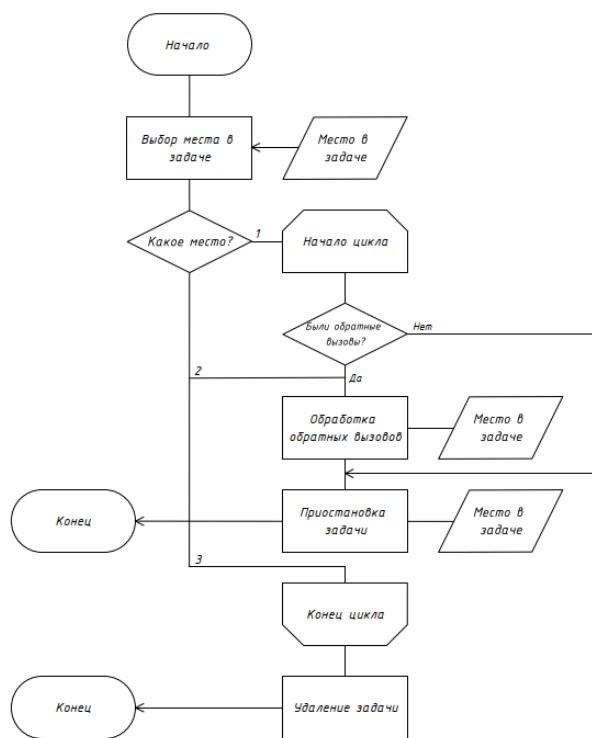


Рис. 2 – Схема алгоритма отправки данных по беспроводной связи

Выполнение представленных алгоритмов в случае безотказной работы не предусматривает выход из цикла, а только приостановку задачи на время. В случае появления ошибок при выполнении, задача завершает цикл выполнения и удаляет задачу.

III. Выводы

Предлагаемая реализация модуля беспроводного информационного взаимодействия с бортовыми системами мобильных машин поможет обеспечить безопасность движения на автомобиле на дорогах общего пользования путём информирования водителя о текущей дорожной обстановке и диагностировать неполадки автомобиля.

Список литературы

1. Espressif API Reference manual [Электронный ресурс]. – Режим доступа: https://docs.espressif.com/projects/espidf/en/stable/esp32/api-reference/network/esp_now.html. Дата доступа: 10.04.2025.

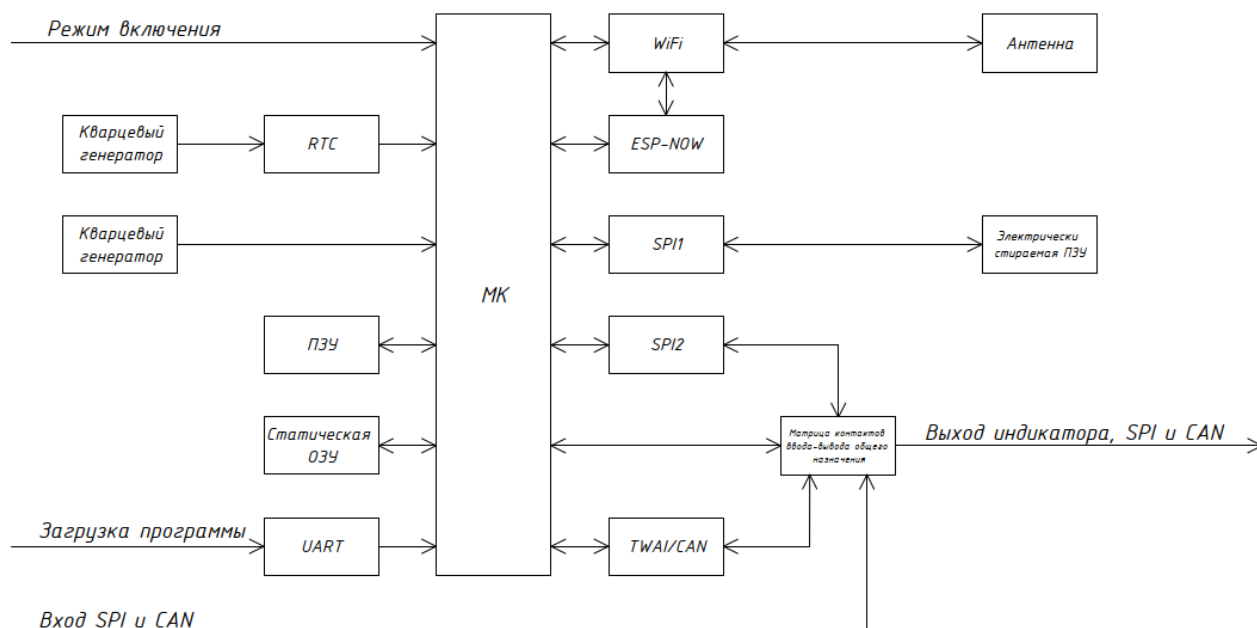


Рис. 3 – Схема структурная модуля беспроводного информационного взаимодействия

Кипцевич Ян Витальевич, студент кафедры теоретических основ электротехники БГУИР, yan.kipcevich@gmail.com.

Научный руководитель: Гоца Виталий Валентинович, младший научный сотрудник ОИМ НАН РБ, mechenistalker@gmail.com.

ПОЛНОМОСТОВОЙ DC-DC ПРЕОБРАЗОВАТЕЛЬ С ФАЗОВЫМ СДВИГОМ

Рассматривается схема реализации полномостового DC-DC преобразователя. Предлагается решение на основе фазового сдвига.

ВВЕДЕНИЕ

Фазовый сдвиг в схеме полного моста играет ключевую роль в обеспечении эффективной и надёжной работы мощных DC-DC преобразователей. Основная задача, которую он решает — реализация переключения при нулевом напряжении, позволяющая существенно снизить потери, возникающие в процессе коммутации МОП транзисторов, особенно при высоких частотах, что позволяет получать высокий КПД при передаче энергии с высоковольтной шины на низковольтную. Схема преобразователя с фазовым сдвигом предоставлена на рисунке 1.

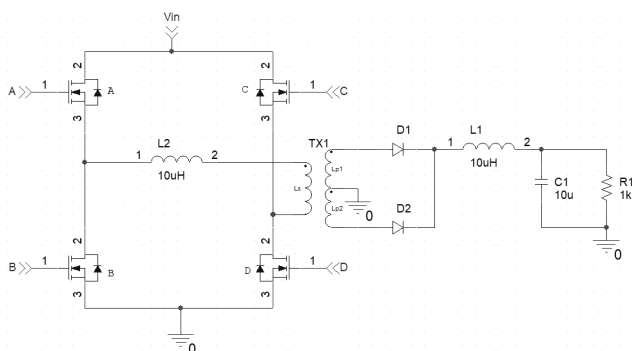


Рис. 1 – Схема PSFB преобразователя

I. ФАЗОВЫЙ СДВИГ

За счёт контролируемого сдвига между управляющими сигналами двух плеч моста создаются условия, при которых энергия, накопленная в паразитной индуктивности (например, рассеянной индуктивности трансформатора), используется для разряда выходной ёмкости выключающегося ключа и зарядки включающегося, тем самым обеспечивая его включение при практически нулевом напряжении. Это не только снижает тепловые потери и улучшает электромагнитную совместимость, но и позволяет работать на более высоких частотах, уменьшая размеры пассивных компонентов и повышая удельную плотность мощности.

Кроме того, фазовый сдвиг позволяет плавно и точно регулировать выходную мощность, делая данную топологию устойчивой и эффективной при разных уровнях нагрузки.

На рисунке 2 показаны управляющие ключами сигналы и их временные смещения относительно друг друга. На рисунке приняты обозначения: T — период управляющего импульса, U_x — напряжение на управляющем контакте соответствующего ключа. Штриховые линии на разных сигналах в один

отрезок времени говорят о передаче энергии в этот отрезок.

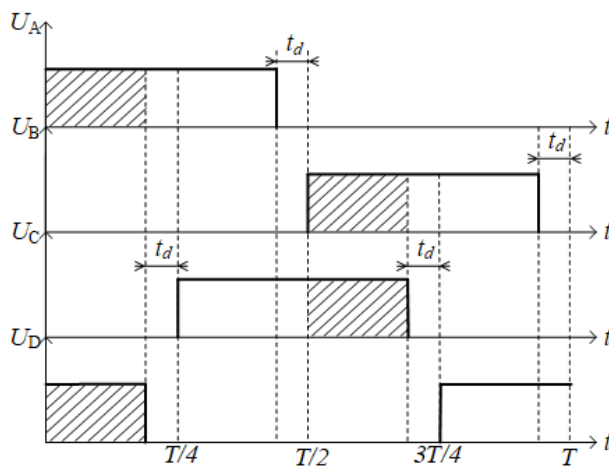


Рис. 2 – Формы сигнала, управляющего ключами

II. УПРАВЛЕНИЕ ФАЗОВЫМ СДВИГОМ

В качестве управляемых ключей используются МОП транзисторы, которые формируют полный мост на стороне первичной обмотки. Диоды D1 и D2 обеспечивают двухполупериодное выпрямление в цепи вторичной обмотки трансформатора, в то время как индуктивность L1 и конденсатор C1 формируют выходной фильтр. Индуктивность L2 помогает утечке индуктивности трансформатора создать резонанс с паразитной ёмкостью МОП-транзисторов, что способствует работе в режиме коммутации при нулевом напряжении.

Транзисторы A и B относительно друг друга управляются сигналом, сдвинутым на 180 градусов, с коэффициентом заполнения импульсов близким к 50 процентам. Поскольку достичь моментального закрытия/открытия транзисторов не возможно, происходит уменьшение коэффициента заполнения на величину t_d , называемым мёртвым временем. Мёртвое время определяется как сумма времён открытия и закрытия ключа. Транзисторы C и D задаются тем же коэффициентом заполнения импульсов и также сдвинуты на 180 градусов.

ШИМ-сигналы управления для плеча A-B сдвинуты по фазе относительно плеча C-D. Величина этого фазового сдвига определяет степень перекрытия работы диагональных ключей, что, в свою очередь, определяет объём передаваемой энергии.

Передача энергии происходит когда открыты транзисторы A и D или B и C. В моменты времени,

когда открыты транзисторы А и С или В и D происходит упомянутый разряд паразитных емкостей.

III. СОКРАЩЕНИЕ ПОТЕРЬ

На рисунке 1 использована двухполупериодная схема выпрямления на диодах. В свою очередь при необходимости получить на выходе схемы достаточно низкое стабильное напряжение, его падение на диодах может стать критичным. Не стоит забывать и про нагрев диодов.

Для уменьшения потерь в цепи выпрямления следует использовать управляемые ключи вместо неуправляемых. Предлагается использовать МОП транзисторы. В такой конфигурации к каждому транзистору добавляется дроссель, которые вместе с выходной ёмкостью образуют сглаживающий фильтр для подавления помех при работе трансформатора. Синхронное управление ключами позволяет в моменты поочерёдной работы катушек разряжать не используемую и тем самым обеспечивает стабильное подавление помех.

IV. ИСТОЧНИК УПРАВЛЯЮЩИХ СИГНАЛОВ

Одной из основных проблем такого преобразователя является управление ключами как силового моста, так и выпрямительной схемы. Из описания работы силовых ключей видно, что сигналы управления сложно объединить между собой. Увеличивать количество контроллеров не рационально, а значит надо находить другое решение данной проблемы. Задача решается применением микросхемы LM5046. Данный ШИМ-контроллер позволяет одновременно управлять всеми ключами, кроме того имеет дополнительный ряд функций. Среди таких функций можно выделить: плавный пуск, стабилизация по току и напряжению, контроль колебаний питающего и входного напряжения преобразователя, защита от скачков напряжения питания, регулировка частоты открытия/закрытия как силовых, так и выпрямительных ключей.

V. МОДЕЛИРОВАНИЕ РАБОТЫ

При изучении работы преобразователя основное внимание уделялось следующим параметрам:

Ламака Евгений Александрович, студент кафедры теоретических основ электротехники БГУИР, lamaka.evgeny@yandex.ru.

Лешиж Захар Иванович, студент кафедры теоретических основ электротехники БГУИР, lpgl2014g@gmail.com.

Научный руководитель: Журавлёв Вадим Игоревич, заведующий кафедрой, кандидат технических наук, vadzh@bsuir.by.

напряжения на вторичной обмотке трансформатора, ток дросселя L1, пульсации в нагрузке, рассеиваемая мощность силовыми и выпрямительными ключами.

При снятии значений с коэффициентом заполнения импульсов 0,47, нагрузке 4 Ом, требуемом выходном напряжении 12 В и токе 3 А, был получен вид напряжения на вторичной обмотке трансформатора, изображённый на рисунке 3.

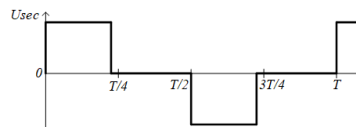


Рис. 3 – Формы сигнала на вторичной обмотке трансформатора при 47% заполнения

В моменты времени, когда напряжение на вторичной обмотке не равно нулю, происходит передача энергии в нагрузку. График тока дросселя имеет пилообразный вид с пиковыми значениями 3,057 и 2,864 соответственно. Размах пульсаций в нагрузке оказался около 0,7 В. На силовых ключах в среднем рассеивается 17 Вт на каждом, а на выпрямительных около 1 Вт суммарно.

VI. ВЫВОДЫ

Предлагаемая топология DC-DC преобразователя хорошо подходит для получения больших мощностей и сокращает потери на ключах в процессе их коммутации. Это позволяет использовать данный преобразователь при передаче энергии с высоковольтной шины на низковольтную.

Список литературы

1. Phase-Shifted Full Bridge DC/DC Power Converter Design Guide / Texas Instruments // Texas Instruments Incorporated . – 2015.
2. Design of Phase Shifted Full-Bridge Converter with Current Doubler Rectifier / Sam Abdel-Rahman // Infineon Technologies North America (IFNA) Corp. – 2013.
3. LM5046 Phase-Shifted Full-Bridge PWM Controller With Integrated MOSFET Drivers / Texas Instruments // Texas Instruments Incorporated – 2024.

МЕТОД ЭФФЕКТИВНОГО ЗАРЯДА НИКЕЛЬ-МЕТАЛЛГИДРИДНЫХ АККУМУЛЯТОРОВ

Рассматривается метод ΔV заряда никель-металлгидридных аккумуляторов и его реализация.

ВВЕДЕНИЕ

Никель-металлгидридные (NiMH) аккумуляторы широко используются в различных областях — от портативной электроники до гибридного транспорта — благодаря своей экологической безопасности, высокой удельной энергии и стойкости к циклическому износу. Однако одним из ключевых факторов, влияющих на срок службы и эффективность этих элементов, остаётся корректный выбор режима зарядки. Неправильный алгоритм зарядки может привести к деградации активных материалов, снижению ёмкости и, в конечном итоге, к преждевременному выходу аккумулятора из строя.

I. ПРОБЛЕМАТИКА ИСПОЛЬЗОВАНИЯ НИКЕЛЬ-МЕТАЛЛГИДРИДНЫХ АККУМУЛЯТОРОВ

Никель-металлгидридные аккумуляторы имеют определённые трудности использования, требующие соблюдения следующих правил для долговечного и безопасного использования:

- Заряжать батарею до полной ёмкости;
- Ограничивать степень перезаряда;
- Избегать высоких температур и чрезмерных температурных колебаний.

Заряд аккумулятора тесно связан с температурой и давлением. Во время заряда положительный электрод достигает полного заряда раньше отрицательного, что может вызвать побочную реакцию гидролиза воды, выделяющую кислород (см. рис. 1). Повышение температуры во время заряда обусловлено экзотермической реакцией рекомбинации кислорода, предотвращающей повышение давления за счёт его реакции с водородом.

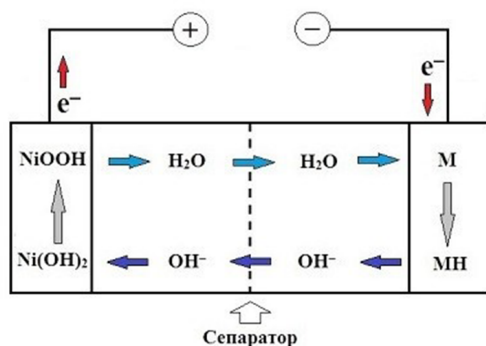


Рис. 1 – Внутреннее устройство никель - металлгидридного аккумулятора

Наиболее распространённым методом зарядки для никель-металлгидридного аккумулятора является заряд постоянным током с ограничением тока.

Это необходимо для предотвращения чрезмерного повышения температуры и давления внутри элемента, а также превышения скорости реакции кислородного восстановления (см. рис. 2).

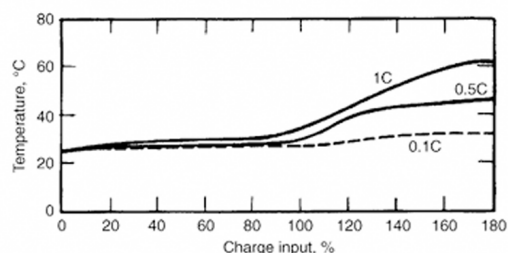


Рис. 2 – График зависимости температуры аккумулятора от величины тока заряда

Характерной особенностью NiMH-аккумуляторов является активное выделение кислорода на положительном электроде при достижении уровня заряда 75–80%. Это сопровождается резким повышением температуры, что приводит к падению напряжения при переходе элемента в стадию перезаряда. Таким образом, напряжение перестаёт быть линейным индикатором степени заряда, что затрудняет точное определение момента достижения полной ёмкости. (см.рис.3.)

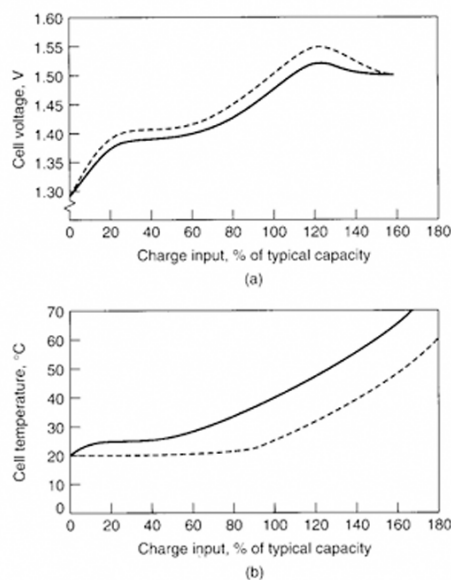


Рис. 3 – Графики напряжения и температуры заряда никель-металлгидридных аккумуляторов

Для решения указанных проблем и более точного контроля процесса зарядки предлагается использовать метод определения момента окончания

заряда на основе критерия отрицательного перепада напряжения (V).

II. МЕТОД DELTA V

При использовании этой методики напряжение во время заряда контролируется, и заряд прекращается, когда напряжение начинает снижаться. Этот подход можно использовать и с никель-металлгидридными батареями, но, пик на металлгидридных элементах не так заметен и может отсутствовать при токах заряда ниже $0,3C$ (где C - ёмкость аккумулятора), особенно при повышенных температурах. Сигнал напряжения должен быть достаточно чувствительным, чтобы прекратить заряд при падении напряжения, но не настолько чувствительным, чтобы преждевременно прекратить заряд из-за шума или других нормальных колебаний напряжения. Для никель-металлгидридных аккумуляторов обычно используется падение напряжения 10 мВ на элемент.

Простейшим способом реализации такого метода зарядки NiMH аккумуляторов является следующее исполнение (см. рис. 4):

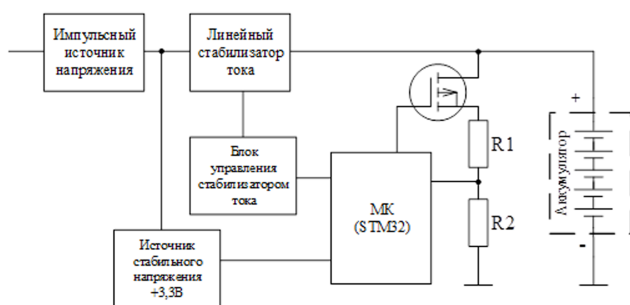


Рис. 4 – Функциональная схема метода заряда дельта V

Функциональная схема разрабатываемого устройства включает в себя следующие ключевые блоки: аккумулятор, делитель напряжения, блок управления на базе микроконтроллера STM32, источник стабильного напряжения на 3,3В для питания микроконтроллера, импульсный источник напряжения, импульсный стабилизатор тока, MOSFET транзистор.

Напряжение с делителя считывается раз в Δt (Используя встроенный в STM32 16 битный таймер 1) на вход 12 битного АЦП STM32 он же пин

ADC максимальное напряжение которого не должно превышать 3,3 - 3,6В.

Далее микроконтроллер считывает значение АЦП входа и сравнивает его с предыдущим. Если это напряжение изменяется на 10-15мВ на элемент, то с выхода пина GPIO микроконтроллера STM32 отправляется управляющий сигнал на отключение заряда аккумулятора. Транзистор MOSFET выступает в роли ключа отключения детекции напряжения с аккумулятора, чтобы исключить саморазряд через делитель напряжения.

Однако для эффективного и безопасного заряда нужно использовать несколько методов, особенно для контроля высокоскоростного заряда. Управление зарядом с помощью термоотсечки может сократить срок службы, поскольку во время заряда батарея обычно нагревается до высоких температур. Этот метод полезен в качестве резервного средства контроля на случай превышения максимальной температуры во время заряда.

Для реализации термоотсечки нужно в нашу схему добавить термодатчик. Термодатчик подключаем ко второму входу 12 битного ADC микроконтроллера STM32. При превышении установленной температуры происходит отправка управляющего сигнала с выхода GPIO микроконтроллера STM32 для прекращения заряда аккумулятора.

III. Выводы

Разработанный метод зарядки с использованием критерия delta V и термоотсечки представляет собой эффективное решение для улучшения процесса зарядки никель-металлгидридных аккумуляторов, обеспечивая оптимальное использование и продление их срока службы. Этот метод способствует повышению эффективности и безопасности использования аккумуляторов, делая их более надёжными и долговечными и предоставляя значительные преимущества для конечных пользователей и промышленности в целом.

1. Linden, D. Handbook of batteries. - 3rd ed. / D. Linden, T. b.Reddy // McGraw-Hill – 1995. – P. 841-912.
2. Zhang, L. Y. Rare Earth-Based Metal Hydrides and NiMH Rechargeable Batteries / L. Y. Zhang // Journal of Alloys and Compounds – Feb 1997.

Смирнов Павел Олегович, студент кафедры теоретических основ электротехники БГУИР, pasha.smirnoff,2004@gmail.com.

Сохин Андрей Андреевич, студент кафедры теоретических основ электротехники БГУИР, soxapro228@gmail.com.

Научный руководитель: Журавлёв Вадим Игоревич, заведующий кафедрой теоретических основ электротехники БГУИР, кандидат технических наук, доцент, vadzh@bsuir.by.

ПРОЕКТИРОВАНИЕ РАЗДЕЛИТЕЛЬНЫХ ФИЛЬТРОВ ДЛЯ МНОГОПОЛОСНОЙ СИСТЕМЫ ВОСПРОИЗВЕДЕНИЯ

Рассматривается раздельный фильтр звуковых сигналов для двухполосной системы воспроизведения.

ВВЕДЕНИЕ

При воспроизведении звука, необходимо разделять входной звуковой сигнал на разные диапазоны частот. Для этого используются многополосные системы. В данной статье будет идти речь о двухполосной системе с фильтрами, построенными по топологии Саллена – Ки.

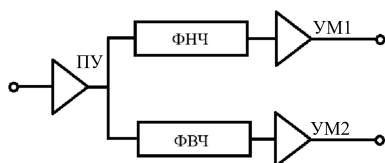


Рис. 1 – Разделение сигнала на нижние и верхние частоты

На рисунке изображена схема двухполосной системы, которая состоит из предварительного усилителя (ПУ), фильтра нижних частот (ФНЧ), фильтра верхних частот (ФВЧ) и усилителей мощности (УМ1 и УМ2).

1. ПРИНЦИП РАБОТЫ ФИЛЬТРОВ

Активные фильтры, построенные по топологии Саллена – Ки, основаны на использовании усилителей с частотно-зависимой обратной связью, формируемой RC-цепями.

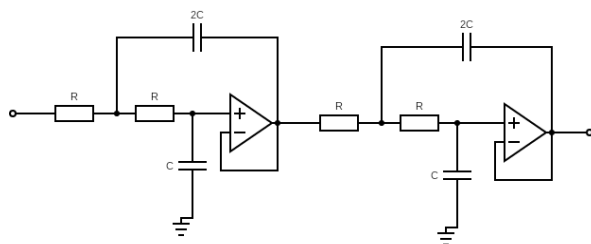


Рис. 2 – Электрическая цепь ФНЧ

Изображённая на рисунке электрическая цепь ФНЧ, используя свойства конденсаторов, позволяет пропускать нижние частоты и отрезать верхние.

Ясинский Павел Михайлович, студент специальности системы управления информацией БГУИР, pavel.yasinsky2006@gmail.com.

Потёмин Иван Викторович, студент специальности системы управления информацией БГУИР, freestyle.06@mail.ru.

Научный руководитель: Шилин Леонид Юрьевич, декан факультета информационных технологий и управления БГУИР, доктор технических наук, профессор, dekfitu@bsuir.by.

Для улучшения качества фильтрации используются два одинаковых каскадно-соединенных участка цепи, которые делают спад АЧХ более крутым.

Отличие ФВЧ от ФНЧ заключается в том, что на месте конденсаторов стоят резисторы, а на месте резисторов – конденсаторы. Соответственно, фильтр пропускает верхние частоты и отсекает нижние.

Теперь необходимо с помощью формулы (1) подобрать такие конденсаторы и резисторы, схема с которыми имела бы частоту пересечения графиков АЧХ обоих фильтров равную, к примеру, 300 Гц.

$$f = \frac{1}{2\sqrt{2}\pi RC}. \quad (1)$$

Возьмём конденсаторы и резисторы, например, с такими характеристиками: $C = 22$ нФ; $R = 18$ кОм. Посчитаем частоту: $f = 284,335$ Гц.

Соединив ФНЧ и ФВЧ, можно получить фильтр, который делит звуковой сигнал на верхние и нижние частоты. Построив АЧХ можно увидеть, что если сложить амплитуды ФНЧ и ФВЧ для той или иной частоты, то получится 1. Это означает, что фильтр работает правильно, так как по итогу должен получиться сигнал с постоянной амплитудой равной 1.

II. Выводы

Таким образом, в работе успешно продемонстрирована фильтрация звукового сигнала на основе его разделения на диапазоны нижних и верхних частот.

1. Бидерман, В. Л. Теория механических колебаний: учеб.-метод. пособие / В. Л. Бидерман, 1980.– 408 с.
2. Марше, Ж. Операционные усилители и их применение: учеб.-метод. пособие / Ж. Марше, 1974.– 118 с.
3. Полонников, Д. Е. Операционные усилители. Принципы построения, теория, схемотехника: учеб.-метод. пособие / Д. Е. Полонников, 1983.– 216 с.

Секция «Гуманитарные дисциплины»

Подсекция «История белорусской государственности»

Председатель:	канд. ист. наук, доцент Николаева Л. В.
Члены жюри:	канд. ист. наук, доцент Литвиновская Ю. И. канд. ист. наук, доцент Куракевич Н. И. канд. ист. наук, доцент Вашкевич И. В. канд. ист. наук, доцент Сугако Н. А.
Секретарь	преподаватель Мякинська А. В.

ИСТОРИЯ РАЗВИТИЯ БЕЛОРУССКОГО ТЕЛЕРАДИОВЕЩАНИЯ

За сравнительно короткий срок телевидение стало одним из важнейших средств идейно-политического, нравственного и эстетического воспитания. Вопросы деятельности телевидения постоянно привлекают внимание исследователей. Необходимость такого исследования назрела, так как за прошедшие десятилетия Белорусское телевидение накопило огромный опыт создания разнообразных общественно-публицистических программ, многие же важные черты телевизионной пропаганды, которые ныне стали ведущими, определились еще в первые годы. Поэтому исследование процесса становления и развития белорусского телевидения поможет постичь законы телетворчества, осмыслить накопленный опыт, сделать важные выводы [1]. Содержание данной работы направлено на формирование информационной культуры и объективного отношения к белорусскому телевидению студентов посредством освещения и трансляции основных направлений истории развития белорусского телерадиовещания до становления в современном телевизионном пространстве. Материалы работы по данной проблеме структурированы в хронологическом порядке. Исследование включает описание основных этапов становления и развития белорусского телерадиовещания, начиная с описания трансляции первой передачи Белорусского радио, которая вышла в эфир 15 ноября 1925 года на радиостанции РВ-10 имени Совнаркома БССР, до развития и становления современного телерадиовещания [2] Работа состоит из четырех разделов, в которых подробно

описаны история создания и развития телерадиовещания на примере Белтелерадиокомпания, а также детально представлена структура Национальная государственная телерадиокомпания Республики Беларусь. Особое внимание уделено изучению и описанию траектории развития Общенационального телевидения: от создания до современности. Рубрика «цитаты, мысли, пожелания» телеведущих Общенационального телевидения дает нам возможность познакомиться с известными телевизионными личностями заочно, узнать интересные моменты из их жизни, о чем они думают, мечтают. В результате работы над проектом была исследована история развития белорусского телерадиовещания с перспективой трансляции данной темы перед общественностью.

Список литературы

1. Архивы Беларуси. История развития белорусского телевидения [Электронный ресурс]. – Режим доступа: <https://archives.gov.by/home/organy-arhivnogo-dela-i-deloizvodstva-i-sistema-gosudarstvennyh-arhivnyh-uchrezhdenij/respublikanskie-arhivnye-uchrezhdeniya/uchrezhdenie-belorusskij-gosudarstvennyj-arhiv-kinofotofonodokumentov-bgakffd/belorusskij-gosudarstvennyj-arhiv-kinofotofonodokumentov/zarozhdenie-televideniya-v-belarusi>
2. ДОСЬЕ: К 60 –летию Белорусского телевидения [Электронный ресурс]. – Режим доступа: <https://belta.by/society/view/dose-k-60-letiju-beloruskogo-televidenija-175706-2015/>

Акуленко И.В., студент 1 курса ФИТУ БГУИР.

Научный руководитель: Мякинська Анна Владимировна, преподаватель кафедры гуманитарных дисциплин

ИСТОРИЯ ТРАНСПЛАНТАЦИИ НА БЕЛАРУСИ

Трансплантация на территории Беларуси прошла долгий путь развития. Началом её истории являются 70-ые годы XX-ого века, когда уролог Николай Савченко провёл первую на территории Беларуси пересадку печени. Ключевым событием в истории трансплантации в нашей стране является первая в независимой Беларуси пересадка печени под руководством Олега Руммо, прошедшая в апреле 2008 года, которая вывела трансплантацию органов в нашей стране на новый уровень. На первых этапах развития пересадки важную роль играло участие Беларуси в составе СССР в программе «Интертрансплант», которая подразумевала установление связей и отношений для эффективной передачи органов и осуществления операций по пересадке органов в социалистических странах. Трансплантация является спорным явлением с точки зрения закона. Было подписано немало актов и документов, регулирующих законодательство в отношении пересадки органов. Вышедшая в 2007 году новая редакция «Закона о трансплантации органов и тканей» стала определяющим и значимым законом, ведь в ней были прописаны основные принципы презумпции согласия, закона о заборе органов на основе согласия со стороны родственников реципиента. С точки зрения морали вставал вопрос о признании смерти умершего, чьи органы, пока не стало поздно, способны подарить жизнь пациенту. Спустя много лет с момента первой успешной трансплантации печени под руководством Руммо Беларусь сохраняет свои высшие позиции в рейтинге стран по пересадке органов человека. Такие успехи – заслуга многих высококвалифицированных специалистов. Некоторые из них стали лицами современной трансплантации Беларуси. Такими можно по праву назвать Олега Руммо, Олега Калачика, Юрия Островского и многих других, которые являлись учениками Николая Савченко или вышеупомянутых специалистов. Под их «крылом» в независимой Беларуси

впервые были проведены операции пересадки печени и сердца. Среди всех ответвлений пересадки органов в нашей стране наибольшего развития удостоились пересадка печени, сердца, легких и костного мозга. Подобные операции являются наиболее доступными, чем в других странах, а их качество ничем не уступает тем, что проводят в Европе. Также важен социальный аспект данного вида вмешательств, ведь для граждан Республики Беларусь все операции по трансплантации органов и тканей проводятся на бесплатной основе. Таким образом, успехи в сфере пересадки органов на территории Беларуси, обеспеченные нашими специалистами и государством, являются предметами для гордости и обосуждений, ведь это – одна из тех областей, которая ставит нашу страну в почётный ряд лидеров мирового здравоохранения. Данная сфера в нашей медицине не стоит на месте, а технологии и техники трансплантации, ничем не уступающие западным, развиваются и совершенствуются до сих пор. В последние десятилетия в республике активно внедряются инновации, которые позволяют минимизировать риск отторжения трансплантированных органов. Не менее важным моментом является совершенствование системы донорства, когда всё большее внимание уделяется как просвещению населения, так и созданию эффективных механизмов для получения органов от доноров. Параллельно с развитием медицинских технологий активно проводятся научные исследования, направленные на создание более совершенных и безопасных методов трансплантации, что открывает новые перспективы для пациентов, нуждающихся в жизненно важных органах. Удивительно, как из области, которую изначально у нас не воспринималась всерьёз, она стала высокоразвитым направлением. Именно поэтому такая сфера, как трансплантация органов и тканей человека, заслуживает большего внимания со стороны правительства и гражданского общества.

Большов И.А., студент 1 курса ФИТУ БГУИР.

Научный руководитель: Мякинська Анна Владимировна, преподаватель кафедры гуманитарных дисциплин

ОТНОШЕНИЕ ЦЕРКВИ, ГОСУДАРСТВА И ОБЩЕСТВА К ВНЕБРАЧНЫМ СВЯЗЯМ В ВЕЛИКОМ КНЯЖЕСТВЕ ЛИТОВСКОМ

Внебрачные отношения в Великом Княжестве Литовском (далее – ВКЛ) регулировались сложной системой писанных и неписанных норм, которые существенно различались в зависимости от сословной принадлежности, гендера и локальных обычаев. Внебрачные связи в ВКЛ были распространенным явлением, несмотря на официальное осуждение со стороны церкви и светского законодательства. Статуты ВКЛ, особенно третий Статут 1588 г., содержали статьи, регулирующие наказания за супружескую неверность, однако применение этих норм существенно зависело от социального положения участников и гендерных аспектов. Для мужчин из привилегированных сословий внебрачные связи редко влекли серьезные юридические последствия. В магнатской и шляхетской среде существовала значительная асимметрия в отношении к мужской и женской супружеской неверности. Связи женатых мужчин с незамужними женщинами более низкого социального статуса часто воспринимались как неизбежное явление и не влекли серьезного общественного осуждения. В то же время измена замужней шляхтянки рассматривалась как тяжкое преступление, угрожающее чистоте наследственной линии и имущественным интересам рода. Статуты ВКЛ предусматривали суровые наказания для неверных жен, вплоть до смертной казни. В крестьянской среде внебрачные связи также не были редкостью, однако их восприятие было более сложным и амбивалентным. Общинная мораль осуждала супружескую неверность, особенно если она становилась публично известной или приводила к семейным конфликтам, нарушающим порядок сельской жизни. В то же время народная культура содержала механизмы примирения и ритуального «очищения», позволявшие сохранить социальный статус участников внебрачных связей, особенно если эко-

номические интересы семьи и общины требовали сохранения брака. Особенно сложной была ситуация с внебрачными детьми. В высших сословиях незаконнорожденные дети находились в юридически уязвимом положении, не имея прав на наследование титулов и родовых имений. Однако в истории ВКЛ известны случаи, когда магнаты и шляхтичи признавали своих внебрачных детей, обеспечивая им образование, содержание и даже добываясь их легитимации через специальные правовые процедуры или королевские привилегии. В крестьянской среде внебрачные дети часто интегрировались в семейную структуру при последующем браке матери, хотя могли подвергаться определенной дискриминации со стороны сельской общины. Церковная юрисдикция в вопросах супружеской неверности дополняла светское законодательство. Епитимьи (церковные наказания) за внебрачные связи включали длительные периоды покаяния, посты и материальные пожертвования на нужды церкви. В многоконфессиональном ВКЛ базовый принцип осуждения внебрачных связей оставался общим для всех христианских конфессий. Таким образом, внебрачные связи в ВКЛ существовали на всех уровнях общества, но их последствия были асимметричны для мужчин и женщин, отражая общее гендерное неравенство эпохи.

Список литературы

1. Сліж, Н. Забароненыя і ненатуральныя сексуальныя стасункі ў ВКЛ у XVI –XVII стст. / Н. Сліж // Вялікае Княства Літоўскае : палітыка, эканоміка, культура : зборнік навуковых артыкулаў. У 2 ч. – Мінск : Беларуская навука, 2017. – Ч. 2. – 2017. – С. 247 –266.
2. Сліж, Н. Шлюбныя і пазашлюбныя стасункі шляхты Вялікага Княства Літоўскага ў XVI –XVII стст. / Н. Сліж. – Смаленск : Інбелкульт, 2015. – 672 с.

Василевский Владислав Валерьевич, студент 1 курса ФИТУ БГУИР

Микша Вячеслав Андреевич, студент 1 курса ФИТУ БГУИР

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

НАСЛЕДИЕ РОДА ПУСЛОВСКИХ КАК ОТРАЖЕНИЕ КУЛЬТУРНЫХ И ИСТОРИЧЕСКИХ ПРОЦЕССОВ XIX ВЕКА

Род Пусловских, принадлежащий к польско-литовской знати, оставил значительный след в истории и культуре XIX века. Их вклад проявился в политической деятельности, развитии экономики, искусства и архитектуры. Представители рода занимали важные посты в Речи Посполитой и Российской империи, активно поддерживали культурные инициативы, предпринимали масштабные экономические проекты. В работе исследуется история рода Пусловских, их архитектурное наследие (дворцы в Коссово и Альбертин, усадьба в Песках), а также влияние семьи на культурные процессы. Отдельное внимание уделено выдающимся личностям, таким как Войцех и Вандалин Пусловские, чья деятельность отражает дух времени. Анализ наследия рода Пусловских позволяет глубже понять исторические и культурные процессы региона в XIX веке, а также роль аристократии в формировании национальной идентичности. Их наследие до сих пор вдохновляет и вызывает интерес, оставаясь важной частью исторической памяти.

Список литературы

1. Архитектурная жемчужина Брестчины: дворец Пусловских в Коссово [Электронный ресурс]. – Режим доступа: <https://travel.by/2024/05/arhitekturnaya-zhemchuzhina-brestchiny-dvorets-puslovskih-v-kossovo/> – Дата доступа: 20.11.2024.
2. Варавва А. Альбертинская усадьба [Электронный ресурс] / А. Варавва. <https://planetabelarus.by/publications/albertinskaya-usadba/>
3. В поисках утраченного. Старинное имение Пусловских «Альбертин» в Слониме на фотографиях начала XX века [Электронный ресурс]. – Режим доступа: <https://www.sb.by/articles/cherty-blagorodnogo-bleska.html>. – Дата доступа: 20.11.2024.
4. Дворец Пусловских в Коссово [Электронный ресурс]. – Режим доступа:
5. Мяснікоў А. Ф. Сто асоб беларускай гісторыі : гістарычныя партрэты / Анатоль Мяснікоў // – Мінск.: Літаратура і Мастацтва : 2009 - 120 с.
6. Памяць: Гіст.-дакум. хроніка Брэста. У 2-х кн. Кн. 1 / Рэдкал. Кісялёў Г. К., Кондак А. П., Куіш М. М., Петрашкевіч А. Л. – Мн.: БЕЛТА, 1997. – 576 с.
7. Памяць: Гіст.-дакум. хроніка Брэста. У 2-х кн. Кн. 2 / Рэдкал. Кісялёў Г. К., Кондак А. П., Куіш М. М., Петрашкевіч А. Л. – Мн.: БЕЛТА, 2001. – 687 с.
8. Род Пусловских [Электронный ресурс]. – Режим доступа: <https://zamok-kossovo.by/history/rod-puslovskikh/> – Дата доступа: 20.11.2024.
9. Усадьба Пусловских в д. Старые Пески [Электронный ресурс]. – Режим доступа: <https://bereza-cbs.by/usadba-puslovskih-v-d-starye-peski/>. – Дата доступа: 20.11.2024.

Волощик Е.Д., студент 1 курса ФИТУ БГУИР.

Научный руководитель: Мякинська Анна Владимировна, преподаватель кафедры гуманитарных дисциплин

ГІСТАРЫЧНАЕ МІНУЛАЕ Ў РАМАНАХ УЛАДЗІМІРА КАРАТКЕВІЧА

Гісторыя цікавіла Караткевіча пераважна як прытча, легенда і ў тым плане, які ўплыў зрабіла яна на духоўнае жыццё народа, на яго грамадскую свядомасць. Пісьменніку характэрна рамантычнае, фальклорна-легендарнае асэнсаванне мінуўшчыны. Паэтызуючы гісторыю, уздымаў пытанні, актуальныя для сённяшняга дня. Уладзімір Караткевіч — пісьменнік з выразна выяўленым лірычна-рамантычным талентам. У гістарычных рэчах ён, ствараючы дасканалыя малюнкi мінуўшчыны, гістарычныя калізіі часцей за ўсё перамяшчае ў сферу рамантычнага канфлікту чалавечнасці, высакароднасці з жорсткасцю, эгаізмам, духоўнай чэрствасцю, цынізмам. Праз творы У. Караткевіча мы далучаемся да гісторыі нашай краіны, у іх прысутнічаюць гістарычныя падзеі розных часоў: ад пачатку стварэння Вялікага Княства Літоўскага да зусім недаўняга мінулага. Героі У. Караткевіча свабодна пераходзяць з аднаго гістарычнага перыяду ў іншы, што абумоўлівае спецыфічную арганізацыю твораў на структурным узроўні. Дзейнічаюць у іншым часе прывідныя коннікі караля Стаха (аповесць «Дзікае паляванне караля Стаха»). Не адпускае мінулае і герояў рамана «Чорны замак Альшанскі». Як трапна заўважае Антон Косміч, той, «хто не памятае мінулага, хто забывае мінулае — асуджаны зноў перажыць яго. Безліч разоў». Мінулае ў разуменні пісьменніка не толькі ўплывае на сучаснае: мінулае як бы працягваецца ў сучаснасці, становіцца яе неад'емнай часткай. Аўтар усведамляе выразную мяжу паміж гэтымі часовымі пластамі, аднак наўмысна паказвае іх злітна. Такое бачанне гісторыі стала наватарскім у беларускай прозе: часова аддалены, працэсуальна завершаны этап мінулага знаходзіць свой працяг у сучаснасці. Праз перакрывацтва розных гістарычных перыядаў Уладзімір Караткевіч паглыблена асэнсоўвае мінулае: кантраснае супастаўленне роз-

ных гістарычных момантаў дазваляе раскрыць новы змест у мінулым, які яшчэ не быў прадметам мастацкага адлюстравання ў рамантычна-асветніцкай традыцыі. Рамантычнае і фантастычнае, прыгодніцкае і дэтэктыўнае ў яго творах не самамэта, не толькі для таго, каб чыталі яго кнігі і на роднай мове. Вывучаючы і даследуючы старонкі гісторыі, У. Караткевіч найперш для сябе зрабіў адкрыццё, што ў нашым мінулым ёсць адказы на пытанні, якія не толькі хваляюць нас сёння, а ад вырашэння якіх залежыць, ці будзем мы «людзьмі звацца». Дзякуючы яго кнігам, мы можам адказаць і на Купалава пытанне, «скуль, якога мы роду». Мы — народ, мы — нацыя не горшая за іншыя, у нас ёсць свае карані, а ў нашым мінулым — не толькі гаротнае, трагічнае, але і гераічнае, вартае гонару і захавання, здрадзіць якому — не мець світання. Нельга не здзіўляцца таму, колькі старонак гісторыі прачытана Уладзімірам Караткевічам для нас, колькі «белых плямаў» зліквідавана, — і ўсё дзеля таго, каб вярнуць нам памяць, бо без памяці нам не быць... Сапраўды, як ён сам казаў, вякі яго спяваюць ротам. Уладзімір Караткевіч стварыў у літаратуры нашу гісторыю, стварыў наша мінулае, з яго лёгкай рукі вякі загаварылі з намі як добрыя субяседнікі, цікава і даходліва. Дык няхай жа гэта размова працягваецца на вякі як памяць аб мастаку рэдкай і непаўторнай сілы, рыцару духу Уладзіміру Караткевічу.

Спісак літаратуры

1. Андраюк С. Беларусь — яго песня // Чырвоная змена. 26 лістап. 1980. С. 4.
2. Верабей А. Абуджаная памяць: Нарыс жыцця і творч. У. Караткевіча. Мінск, 1997. 256 с., [8] л. іл
3. Караткевіч У. Збор твораў у васьмі тамах. Мінск, 1987–1991.
4. Ліс А. С. Годна, з любасцю // Ліс А. С. Цяжкая дарога свабоды. Мінск, 1994. С. 394–413.

Гаркуша А.С., студэнтка 1 курса ФІТіК БДУІР.

Данілава К.В., студэнтка 1 курса ФІТіК БДУІР.

Навуковы кіраўнік: Мякінская А.У., выкладчык кафедры гуманітарных дысцыплін

ЖИЗНЕННЫЙ ПУТЬ РУДЕНКОВА НИКОЛАЯ ФЁДОРОВИЧА И РУДЕНКОВОЙ КЛАВДИИ ФЕДОТОВНЫ

Мой прадед Руденков Николай Фёдорович родился 20 февраля 1917 года в деревне Зябровка Гомельского района. Закончил 4 класса школы. Трудился в колхозе до момента призыва в ряды РККА. В октябре 1938 года призван в железнодорожные войска во 2-й железнодорожный полк в качестве курсанта-мостовика, где 20 февраля 1939 года принял воинскую присягу. После прохождения курсов в сентябре 1939 года зачислен в батальон связи 2-го железнодорожного полка в качестве механика связи. Войну встретил в возрасте 24 лет в звании сержанта. Был направлен на Южный фронт. В июле 1941 года на переправе через реку Днестр его подразделению была поставлена задача: заминировать, а после отхода подразделений Красной армии взорвать мост через реку. Однако передовые силы противника вышли к данному мосту раньше отступавших частей. Он в составе отряда удерживал мост в течение 2 часов. Отряд понес серьезные потери, сам же Николай Фёдорович был ранен и проходил лечение в Одесском госпитале. За этот бой он был представлен к награде: ордену Красной Звезды. После эвакуации войск с Одессы он с полком был направлен на Донбасс. В ноябре 1941 года в звании старшего сержанта зачислен во 2-й восстановительный железнодорожный батальон на должность старшины. Далее его батальон дислоцировался в таких местах, как Кавказ, Курщина, Смоленщина, Беларусь, Польша, Германия. За годы войны был награжден медалями: “За оборону Одессы”, “За оборону Кавказа”, “За боевые заслуги”, “За освобождение Варшавы”, “За взятие Берлина”, “За победу над Германией”. Указом Президиума Верховного Совета СССР от 11 марта 1985 года был награжден орденом Отечественной Войны 2 степени. Ершова Клавдия Федотовна родилась 18 января 1925 года в деревне Степанищево Тамбовской области. В семье она была старшей из 4-х детей. Закончила 8 классов школы.

Войну она встретила в возрасте 16 лет. В сентябре 1941 года её отец был призван на фронт. Он был пленен в октябре 1941 года и отправлен в концлагерь. В 1942 году при побеге одну из групп, в которой он находился, настигли немцы, расстреляв всех на месте. Узнав от сумевших сбежать лагерников про смерть отца она поставила цель: любым способом приблизить победу. В осенью 1942 года в возрасте 17-ти лет вместе с 2-мя подругами сбежала из дома в город, добираясь на подножке поезда. Разыскав военкомат и приписав себе 1 год к возрасту, она была направлена на курсы радисток, которые закончила в 1943 году, и отправлена на фронт. В этом же году познакомилась с моим прадедом – Руденковым Николаем Фёдоровичем, с которым бок о бок и прошли всю войну. Самым ярким воспоминанием прабабушки является посещение Георгием Константиновичем Жуковым пункта связи в 1944 году, на котором находилась и она. Георгий Константинович поблагодарил связисток за службу и сказал: “Доченьки, потерпите. Война скоро закончится”. После взятия Берлина прабабушка и прадедушка распались на рейхстаге. Вскоре они заключили брак. Клавдия демобилизовалась летом 1945 и уехала домой. Руденкова Клавдия Федотовна была награждена медалью “За победу над Германией”. Указом Президиума Верховного Совета СССР то 11 марта 1985 года была награждена орденом Отечественной Войны 2 степени. После демобилизации прадед, забрав прабабушку из деревни, поселился в Гомеле. После войны прадед работал на руководящих снабженческих должностях. Прабабушка же изначально работала продавцом, а позднее телефонисткой. У них родилось четверо детей, из которых выжили трое. Умер Руденков Николай Фёдорович 9 сентября 1993 года. Руденкова Клавдия Федотовна умерла 13 августа 1997 года.

Забаровский Артём Алексеевич, студент 1 курса ФИТУ БГУИР.

Научный руководитель: Качалов Игорь Леонидович, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

РАСКОПКИ НА Р. МЕНКЕ: НОВЫЕ ОТКРЫТИЯ

Одной из важнейших задач археологических исследований на р. Менке было определить, здесь ли на самом деле зародился Минск. Город впервые упоминается в летописи «Повесть временных лет» в связи с битвой на реке Немиге, состоявшейся в 6575 г. от сотворения мира, или в 1067 г. от Рождества Христова. Где же тогда находился Менск? Ранее археологами было установлено, что в 17 км от минского замчища в деревне Городище Щомыслицкого сельсовета расположено крупное городище. Обнаруженные в нем находки позволяли говорить, что там, на реке Менке и ручье Дунае, находилось поселение, подходящее по возрасту под летописный Менск. Второй из важнейших задач раскопок на Менке было определение возраста оборонительных укреплений вокруг городища. Результаты исследований показали: укрепления построили в конце X – начале XI в. Теперь можно говорить смело – городище на Менке точно существовало в 1067 г. В отличие от Минска на Свислочи. Менск окружали могучие деревянные стены высотой 5 м. Таким образом, на то время это единственное серьезно укрепленное поселение в округе – настоящий город. Деревянные срубы «городни», из которых состояла стена, недолговечны. Пошатнувшиеся конструкции для придания устойчивости засыпали землей у основания и надстраивали. Так с течением десятилетий стены превратились в валы. На территории всей Восточной Европы подобные конструкции того периода археологами не фиксировались. Это был крупный населенный пункт, первоначальное ядро населения которого составили славяне, пришедшие с южных земель в IX в. Далее община освоила это место, оно постепенно переросло в городской центр. Здесь проводились крупные торговые сделки, была ювелирная мастерская и в какой-то момент появилась княжеская администрация. При раскопках вала археологи нашли третьи въездные ворота на территорию городища. Исследования показали, что они соору-

жены во времена ВКЛ. Все это удалось узнать в результате исследований 2023 г. Последние археологические исследования части минского замчища около современной пл. 8 Марта, а также ремесленного и торгового центра в районе нынешних ул. М. Богдановича, Зыбицкой, Кирилла и Мефодия, Немиги, Интернациональной дали богатый материал. Только находок, датированных ранее второй половиной XI в., там не обнаружено. Этот факт – одно из доказательств того, что укрепления на Свислочи возвели после летописных событий. Благодаря масштабным раскопкам на реке Менке появилась возможность по-новому взглянуть на жизнь наших предков в X в. Здесь обнаружены уникальные артефакты. Например, кистень X–XI вв. – бронзовое грушевидное ударное оружие ближнего боя. Или семь деревянных лопат, изготовленных более тысячи лет назад. Целая коллекция торговых пломб указывает на широкую торговлю этого региона с отдаленными землями. А книжная застёжка говорит о наличии книг и грамотности местных жителей, что для X – первой половины XI в. очень редкая вещь! Выявлены и бесспорные доказательства присутствия в поселении скандинавов. Но самые яркие находки донца с изображением княжеского трезубца князя Изяслава, сына Рогнеды и Владимира. Первое такое найдено в 2022 г., второе в 2023 г. Это очень важный предмет как для истории Минска, так и для всей Беларуси. Ведь донце представляет собой символ княжеской власти и, по сути, служит отправным пунктом в развитии государственности на наших землях. Таким образом, комплекс на Менке ярко иллюстрирует каноничность и историчность государствообразующих процессов на белорусской земле.

Список литературы

1. Войтехович, А. В. Предварительный отчет о результатах археологических и исторических исследований археологического комплекса на р. Менке в 2023 году / А. В. Войтехович. – Минск : б. и., 2023. – 39 с.

Зубович Егор Дмитриевич, студент 1 курса ФИТУ БГУИР.

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

ВИЛЕНСКО-РАДОМСКАЯ УНИЯ 1401 Г.: ИТОГИ И ПОСЛЕДСТВИЯ

Виленско-Радомская уния 1401 г. стала ключевым этапом в развитии польско-литовских отношений, закрепив и модифицировав положения Кревской унии 1385 г. Она создала новую модель двусторонних отношений. Одним из итогов унии стала пожизненная передача великокняжеской власти в ВКЛ князю Витовту, что означало признание Польшей государственной независимости ВКЛ. Важным являлось отсутствие термина «инкорпорация» и замена его на «включение», «присоединение». Эта формулировка давала возможность интерпретации и создавала компромисс между сторонами. Вторым итогом унии являлось урегулирование вопроса выборов польского короля после бездетной смерти Владислава II Ягайло. В этом случае поляки должны были выбирать короля с «уведомлением и советом» литовцев. Роль литовцев должна была быть небольшой, но создавалась возможность для сотрудничества между обеими сторонами. Третий итог – военный союз. Обе стороны заявили, что в случае нападения иностранного государства на одну из них другая сторона окажет помощь атакованной стороне. В данном случае обязательства были двусторонними и равнозначными. Четвертым итогом унии было участие общественного фактора. Впервые договор между правителями двух стран был подтвержден обществом. Особенно важным было привлечение литовских панов и бояр к соучастию в управлении. Соглашение стало делом двух народов. Последний итог касался разделения власти между Владиславом II Ягайло и Витовтом. У Витовта была полная великокняжеская власть во внутренней политике и ограниченная во внешней. Витовт должен был признать над собой верховенство польского короля из династии Ягайло и польского государства. Последствия унии оказались значительными. Были урегулированы польско-литовские отношения по принципу двустороннего компромисса.

Сущность его сводилась к примирению двух позиций: государственной независимости ВКЛ и принадлежности ВКЛ к польскому государству. ВКЛ было юридически восстановлено. Витовт получил полную власть ВКЛ. С одной стороны, это успокаивало амбиции Витовта, а с другой – стремление литовских бояр иметь государство, которым не управляли бы чужеземцы. Сложилась основа для тесного сотрудничества между Владиславом II Ягайло и Витовтом на основе четкого разделения компетенций. Произошло политическое усиление литовских панов и бояр путем участия их в заключении унии и объявление ограниченного участия в будущей элекции польского короля. Наблюдалось ухудшение литовско-тевтонских отношений и фактическая денонсация Салинских договоренностей 1398 г. Витовтом. Международное положение монархии Владислава II Ягайло усилилось. Договором был недоволен князь Свидригайло, так как он лишился надежд на великокняжескую власть в ВКЛ. Поэтому он немедленно начал организовывать восстание, заручившись поддержкой Тевтонского ордена (что вполне объяснимо), Мазовии (здесь у него были родственники) и русских князей. Таким образом, Виленско-Радомская уния не просто урегулировала текущие противоречия, но и создала правовую основу для дальнейшего развития польско-литовских отношений, существенно повлияв на политическую карту Восточной Европы начала XV в.

Список литературы

1. Błaszczyk, G. Dzieje stosunków polsko-litewskich: w 2 t. / G. Błaszczyk. – Poznań : Wydawnictwo Naukowe Uniwersytetu im. A. Mickiewicza. – Wydawnictwo Poznańskie, 1998 – 2007. – Т. II. Od Krewa do Lublina. – Część I. – 2007. – 936 s. – S. 249–283.
2. Halecki, O. Dzieje Unii Jagiellońskiej / O. Halecki. – Kraków : Akademia Umiejętności, 1919/1920. – Т. 1. – 1919. – 482 s.

Новиков Денис Александрович, студент 1 курса ФИТУ БГУИР.

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

РАЗВИТИЕ ЖЕЛЕЗНОЙ ДОРОГИ РЕСПУБЛИКИ БЕЛАРУСЬ

Появление железной дороги сыграло большую роль в истории. Созданный в XVIII веке паровой двигатель стал началом развития данной отрасли. А с 1862 года по наши дни железнодорожный транспорт является неотъемлемой частью истории нашей страны. Густая сеть железных дорог, опоясавших Беларусь, содействовала ускорению темпов промышленного развития страны на всех ее этапах развития. В революционное время, в годы первой и второй мировых войн железная дорога так же играла большую роль. Сейчас железнодорожные пути используются для перевозки грузов, обеспечивают транспортную инфраструктуру, которая поддерживает различные отрасли экономики, такие как сельское хозяйство, добыча полезных ископаемых и промышленное производство. Они позволяют перево-

зить сырье и готовую продукцию по всей стране, за границу и из-за границы, обеспечивая доступность и конкурентоспособность товаров на рынке. Развитие железной дороги несомненно сыграло большую роль в истории. Появление нового транспорта существенно ускоряло развитие городов и стран, упрощало транспортировку больших грузов, а также стало одним из самых доступных и быстрых способов передвижения людей, в том числе простого населения, между городами и странами. Немаловажную роль железная дорога играла и во время войн и революций, став важным стратегическим объектом. Сегодня развитие данной отрасли не останавливается и набирает новые обороты. До сих пор перевозка грузов на поездах является самым дешевым способом транспортировки на суше.

Павлюченко Ю. М., студент 1 курса ФИТУ БГУИР.

Научный руководитель: Мякинська Анна Владимировна, преподаватель кафедры гуманитарных дисциплин

ЧЕРНОБЫЛЬСКАЯ КАТАСТРОФА: БЕЛОРУССКИЙ ОПЫТ ПРЕОДОЛЕНИЯ ПОСЛЕДСТВИЙ И УРОКИ ДЛЯ МИРА

Авария на Чернобыльской атомной электростанции (ЧАЭС) 26 апреля 1986 года представляет собой крупнейшую техногенную катастрофу в истории атомной энергетики. Республика Беларусь оказалась наиболее пострадавшей страной: около 23% ее территории подверглось радиоактивному загрязнению, преимущественно долгоживущими радионуклидами. Масштаб и долгосрочный характер последствий обусловили необходимость разработки и реализации беспрецедентных по объему и продолжительности мер по их преодолению. Анализ опыта пострадавших стран и, в частности, белорусского опыта, накопленного за десятилетия после аварии, представляет большой научный и практический интерес. Систематизация и анализ ключевых моментов этого опыта преодоления последствий Чернобыльской катастрофы и формулирование на его основе уроков, имеющих значение для обеспечения радиационной безопасности и управления рисками техногенных катастроф на международном уровне, а также рассмотрение масштабов радиоактивного загрязнения территории Беларуси, основных направлений государственных контрмер в социально-экономической, медицинской и аграрной сферах способствует укреплению безопасности в условиях интенсивного развития атомной энергетики на всех континентах земного шара. В результате аварии на ЧАЭС на территорию Беларуси выпало, по различным оценкам, около 70% всего объема радиоактивных веществ, выброшенных из разрушенного реактора. Основное загрязнение сформировано цезием-137 (Cs-137), стронцием-90 (Sr-90) и изотопами плутония (Pu-238, Pu-239, Pu-240), обладающими длительными периодами полураспада и представляющими долгосрочную угрозу для окружающей среды и здоровья населения. Наиболее интенсивному загрязнению подверглись Гомельская и Могилевская области. Площадь сельскохозяйственных угодий, загрязненных цезием-137 с плотностью выше 37 кБк/м² (1 Ки/км²), составила более 1,8 млн гектаров, а площадь лесов – около 2 млн гектаров. Вследствие установленных уровней загрязнения и дозовых нагрузок из наиболее пострадавших районов было эвакуировано и переселено около 138 тысяч человек. Значительное число жителей (более 1,5 миллиона человек на первоначальном этапе) продолжали проживать на территориях с различной степенью радиоактивного загрязнения, что потребовало со стороны властей внедрения комплексной системы радиационной защиты и мониторинга. Реакция Республики Беларусь на катастрофу носила системный и долгосрочный характер. Она реализовалась через принятие специальных законов (такие, как "О социальной защите граждан, пострадавших от катастрофы на Чернобыльской АЭС

"О правовом режиме территорий, подвергшихся радиоактивному загрязнению в результате катастрофы на Чернобыльской АЭС") и ряда государственных программ по преодолению последствий. Основные меры включали создание системы постоянного радиационного контроля окружающей среды, продуктов питания, сельскохозяйственной продукции; дозиметрического контроля населения; установление допустимых уровней содержания радионуклидов в продуктах питания и питьевой воде; разработку и широкомасштабное внедрение защитных мероприятий в сельском хозяйстве (известкование кислых почв, применение повышенных доз калийных и фосфорных удобрений, подбор культур с низким коэффициентом накопления радионуклидов, переспециализация хозяйств) для снижения попадания радионуклидов в продукцию растениеводства и животноводства; предоставление льгот и компенсаций пострадавшему населению (включая медицинское обслуживание, санаторно-курортное лечение, пособия); реализацию программ по социально-экономическому развитию пострадавших регионов с целью создания безопасных условий проживания и трудовой деятельности; организацию системы диспансеризации и специализированной медицинской помощи населению, проживающему на загрязненных территориях; создание Национального регистра лиц, подвергшихся воздействию радиации. Особое внимание уделялось мониторингу и лечению рака щитовидной железы у детей и подростков, так как наблюдался интенсивный рост заболеваемости. Этот вид рака был признан ВОЗ и UNSCEAR прямым медицинским следствием аварии. Рассмотрение опыта Беларуси в преодолении последствий Чернобыльской катастрофы позволяет сформулировать ряд фундаментальных уроков, имеющих существенное значение не только для пострадавших на тот момент стран, но и для всего мирового сообщества. Катастрофа продемонстрировала критическую важность внедрения и поддержания высокого уровня культуры безопасности на всех этапах жизненного цикла ядерных установок – от проектирования до вывода из эксплуатации. Это включает и строгие нормативные требования, и независимый надзор, и постоянное обучение персонала и формирование у него ответственности за безопасность, как это подчеркивается в стандартах МАГАТЭ. Недостаток и искажение информации в первые недели после аварии значительно усугубили ее последствия и подорвали доверие населения, поэтому потребовалась политика максимальной информационной открытости, оперативного предоставления населению достоверных данных о радиационной обстановке, потенциальных рисках и мерах защиты. Последствия крупномасштабной катастрофы выходи-

ли далеко за рамки непосредственного физического ущерба, поэтому была необходима и долгосрочная государственная поддержка, включающая не только медицинскую помощь и материальные компенсации, но и программы психологической реабилитации, социальную адаптацию переселенцев и работу по преодолению стигматизации пострадавших территорий и их жителей. Преодоление последствий требует разработки и последовательной реализации долгосрочных, научно обоснованных стратегий реабилитации загрязненных территорий, и это включает не только технические аспекты (как, например, дезактивация, агротехнические меры), но и создание условий, необходимых для безопасного проживания и устойчивого социально-экономического развития этих регионов и интеграцию их в национальную экономику на новых принципах. Масштаб Чернобыльской катастрофы и ее трансграничный характер подчеркнули необходимость тесного международного сотрудничества в области ядерной и радиационной безопасности, обмена информацией, научными исследованиями и опытом преодоления последствий аварий. Международные организации, такие как МАГАТЭ, ВОЗ, ПРООН, играют в этом ключевую роль. Чернобыльская катастрофа оказала сильнейшее и долговременное влияние на Республику Беларусь. Накопленный за 38 лет страной опыт преодоления ее последствий является уникальным и многогранным. Приоритет безопасности,

важность прозрачности и открытости в информировании населения, предоставление комплексной социальной поддержки, стратегическое планирование реабилитации и необходимость тесного международного сотрудничества стали ключевыми уроками для всего мира. Они должны быть усвоены и интегрированы в национальные и международные системы управления рисками для предотвращения повторения подобных трагедий и эффективного и своевременного реагирования на них в будущем.

Список литературы

1. Беларусь и Чернобыль: 35 лет спустя. Информационно-аналитические материалы. / «Белорусское отделение Российско-белорусского информационного центра по проблемам последствий катастрофы на Чернобыльской АЭС» Научно-исследовательского института пожарной безопасности и проблем чрезвычайных ситуаций МЧС Республики Беларусь. – Минск : Беларусь, 2021. – 49 с.
2. О социальной защите граждан, пострадавших от катастрофы на Чернобыльской АЭС, других радиационных аварий : Закон Респ. Беларусь от 6 января 2009 г. № 9 – З // *ilex* : информ. правовая система (дата обращения: 11.04.2025).
3. О правовом режиме территорий, подвергшихся радиоактивному загрязнению в результате катастрофы на Чернобыльской АЭС : Закон Респ. Беларусь от 11 ноября 2019 г. № 252 –З // *ilex* : информ. правовая система (дата обращения: 11.04.2025).
4. Международное агентство по атомной энергетике [Электронный ресурс]. – Режим доступа: <https://www.iaea.org>. – Дата доступа: 11.04.2025.

Тетёркина С.П., студентка 1 курса ФИТУ БГУИР.

Научный руководитель: Сугако Наталья Анатольевна, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

ЖИЗНЕННЫЙ ПУТЬ И ВОСПОМИНАНИЯ О ВЕТЕРАНЕ ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ ПИРОГОВСКОМ Е. В.

Данная работа посвящена герою Великой Отечественной войны, Пироговскому Евгению Ивановичу, который по совместительству является моим прадедом. Евгений Иванович является примером героизма, проявленного советскими гражданами в тяжелые времена Великой Отечественной войны. В работе представлена биография Евгения Ивановича, составленная из документов архива «Память народа», и различных историй из жизни Пироговского, рассказанных им лично или его дочерью, моей бабушкой, Жуковой Светланой Евгеньевной. Эти истории показывают военные действия со стороны обычного человека, который готов на любые поступки ради защиты своей страны. Также в работе рассказывается о послевоенном периоде и о том, как бывший военный способен освоиться в мирное время. Данная работа актуальна в преддверье праздника Великой Победы. Из поколения в поколение, как святыня, передается память предков. Моя семья бережно хранила благодарственное письмо от И.В. Сталина моему прадеду нефтянику Евгению Ивановичу Пироговскому. Пожелтевший от времени документ свидетельствует о мужестве и отваге танкиста Пироговского. Евгений Иванович Пироговский – родился 28 марта 1921 года в селе Бородилы Могилевской области. Герой Великой Отечественной войны, трижды награжден медалью «За отвагу» и орденом Великой Отечественной войны. Почетный работник предприятия «Белоруснефть». Умер 11 декабря 2013 года в Гомеле. За фронтовые годы Солдату Победы была вручена двадцать одна благодарность. Среди них — за освобождение от фашистских захватчиков Белгорода, Харькова, Будапешта, Братиславы, Вены и Праги, а также за форсирование Хинганского хребта в Маньчжурии. 20-летним юношей Евгений Пироговский, как и многие его сверстники, попал на войну. И первое боевое крещение он получил... в бане. С фронта грязь смыть не удалось – прямое попадание снаряда уничтожило почти всех, кто там находился. «Невозможно ощутить ужас, страх и боль, когда наших товарищей, которые несколько минут назад шутили и смеялись, пришлось собрать в окровавленный плащ-палатку», - говорит Евгений Иванович. «Руки у меня дрожали, разум терзался от увиденного». А через несколько месяцев он пережил еще одну кровавую «ванну», под Ростовом. Там он впервые осознал величину жертвы, которую пришлось принести за каждую завоеванную деревню. Через прицел семидесятимиллиметровой противотанковой пушки он видел только позиции противника, не замечая того, что происходило вокруг. И когда в чудом уцелевшем аппарате остались в живых только два бойца, которые как-то не чувствовали страха в горле. Напротив, какое-

то отчаянное безрассудство, дикая страсть овладели им. «Пока я жив и все оружие живо – фашистам здесь не место. Эта мысль стучала в моей голове, как молот», — вспоминает Пироговский. Но судьба недолго берегла юную бойца. Его противотанковый расчет был уничтожен под Таганрогом. Сам он был тяжело ранен и выздоровел лишь через три дня на медицинском судне. Состояние его было критическим, и Евгений понимал, что даже если он выздоровеет, то, вероятно, будет признан негодным к дальнейшей службе. Больницы, врачебные комиссии... Вердикт врачей - негоден. Пироговский отправляется к сестре, которая, собственно, и поставила его на ноги. Потому что она увидела: брат рядом, и его душа там, на передовой. Узнав, что его родной полк переведен в Нагинский, он быстро собрал сумку и отправился на станцию... Их расчет был брошен на прорыв под Харьковом. Евгений Пироговский, уткнувшись лбом в прицел, смотрел на раскинувшееся впереди поле, перепаханное танковыми гусеницами, и буквально оцепенел: танки, как огромные жуки, двигались вниз, превращаясь в турку, полусладкие черные тучи курил, полз. ...Рёв моторов был такой, что казалось, наступит минута молчания, и ты оглохнешь. После перерыва начались долгожданные победы. Были отбиты Белгород, Кировоград, Умань, Молдавия. Затем боевые действия переместились на Румынию, Австрию и Венгрию. Евгений Пироговский вспоминает, как в Чехословакии он мчался на танке по разрушенным улицам, чтобы спасти маленькую раненую девочку, неизвестно, как она оказалась в расположении их части. Но многострадальная Победа не означала для Евгения конца войны. В продуваемых всеми ветрами теплицах он и его боевые товарищи были отправлены в безводные монгольские степи. Другие люди, незнакомые русскому глазу пейзажи, труднопроизносимые названия городов: Хайлар, Цыкар, Бинсиху, Мукден и как следствие медаль «За победу над Японией». За проявленную храбрость Евгений Иванович был награжден благодарственным письмом Иосифа Сталина. За боевые подвиги Евгений Иванович награжден 3 медалями за отвагу (1943, 1944, 1945 гг.) и орденом Отечественной войны первой степени (1985 г.). Эти и многие другие медали хранятся в моей семье и к ним относятся с особым уважением. Первая награда: за то, что в период боев на фронте, несмотря на тяжелые условия работы и сильный огонь противника, приняли меры по улучшению работы штаба и безопасности личного состава полка. Работа поставлена хорошо, налажена доставка послания семьям погибших воинов. Вторая награда: за то, что, работая старшим делопроизводителем, т. Пироговский полностью от-

дался своему делу. При обороне в районе города Яса они многое сделали в личном плане. Все бухгалтерские книги ведутся согласно приказу. В ходе прорыва обороны противника и подготовки к прорыву он много работал на оперативном поприще, что позволяло быстро разрабатывать планы наступления. Третья награда: за то, что своим неустанным трудом обеспечил техническую работу штаба. Во время боевого марша он принимал активное участие в строительстве переправ, своей энергичной работой вдохновлял солдат на выполнение боевого приказа. В ходе операции он неустанно выполнял все поставленные перед ним задачи, своевременно оформляя документы, что давало возможность штабу управлять частями. Вернувшись с Дальнего Востока, Евгений, как мог, начал обживаться и обустриваться в мирной жизни. Он поселился в Гомеле, женился, вырастил троих детей. С 1966 года началась его работа в объединении «Белоруснефть». Двадцатилетний трудовой стаж был посвящен работе в отделе мобилизационного отряда, который он возглавлял. Вот тут-то и пригодились его военная подготовка, дисциплина и ответственность. Бывшие коллеги хорошо помнят его за простоту в общении, скромность и постоянную порядочность Евгений Иванович был строгим, но очень веселым и активным человеком. У него было много знакомых и друзей, которым он

всегда помогал. Бывало, он помогал даже совершенно незнакомым людям. Между моим прадедом и Андреем Громыко установлена родственная связь. Они дальние родственники. Также они лично были знакомы и порой пересекались как по личным, так и по рабочим вопросам. Большую часть информации мне рассказала моя бабушка по материнской линии, дочь Евгения Ивановича Светлана Евгеньевна. Благодаря ей я многое знаю об истории своей семьи. Я очень благодарен ей за помощь и рассказы многих интересных историй. Евгений Иванович умер 11 декабря 2013 года. Мне очень повезло, и я застал его живым. К сожалению, из-за малого возраста я помню немного, но точно знаю, что навсегда запомню одно: его добрые глаза, отражающие не только мощный характер, выкованный войной, но и настоящую силу: любовь к своим родственникам и своей родине. Я очень горжусь своим прадедом и осознаю, что именно благодаря ему и миллионам других героев я сейчас живу в мире и спокойствии. Евгений Иванович – представитель поколения победителей, людей, рискнувших всем ради светлого будущего для своей страны и своих близких. Как капля в море, как песчинка в пустыне, моя семья – одна из миллионов страниц очень интересной и увлекательной книги под названием «История жизни».

Федорин Тимофей Леонидович, студент 1 курса ФИТУ БГУИР.

Научный руководитель: Качалов Игорь Леонидович, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

СОЗДАНИЕ ЗАПАДНОЙ ОБЛАСТИ – ЗАПАДНОЙ КОММУНЫ КАК ПРЕДШЕСТВЕННИЦЫ ССРБ

Западная область – административно-территориальная единица в Российской республике и в Советской России, образованная Всероссийским совещанием Советов в марте 1917 г. В состав Западной области изначально входили Витебская, Могилевская и Минская губернии, неоккупированная часть Виленской губернии, а с апреля 1918 г. также Смоленская губерния. Административным центром Западной области стал Минск, где располагался штаб Западного фронта, позднее – Смоленск. После Февральской революции 1917 г. военные гарнизоны в западных губерниях становились политическими центрами, в которых солдаты становились ведущей силой. Западный фронт превратился в политический центр всего региона. К осени 1917 г. война, разруха и надвигающийся голод привели к росту популярности левых (социалистических) идей. 7 ноября (25 октября) 1917 г. в Петрограде в результате вооруженного восстания было свергнуто Временное правительство и власть перешла в руки Советов рабочих, солдатских и крестьянских депутатов, в которых преобладали представители РСДРП(б). После победы Октябрьской революции Минский совет рабочих и солдатских депутатов 25 октября (7 ноября) 1917 г. взял власть в городе. 27 октября (9 ноября) при нем был создан Военно-революционный комитет Западной области и фронта (ВРК). Фактическое формирование Западной области было завершено 26 ноября (9 декабря) 1917 г. созданием объединенного Исполкома Советов рабочих, солдатских и крестьянских депутатов и Совнаркома Западной области и фронта (далее – Облисполком-зап, председатель Президиума – Н.В. Рогозинский). Для координации деятельности отделов Облисполком-запа был учрежден Совет Народных Комиссаров Западной области и фронта (далее – СНК, председатель и комиссар внутренних дел – К.И. Ландер). В связи с оккупацией большей части территории

Беларуси немецкими войсками в феврале 1918 г., закрепленной 3 марта 1918 г. Брестским миром, центр Западной области был перенесен из Минска в Смоленск. Согласно постановлению II съезда Советов Западной области, проходившего в апреле 1918 г., в ее состав была включена Смоленская губерния. 11–13 сентября 1918 г. на III съезде Советов Западная область была переименована в Западную коммуну с центром в Смоленске. После революции 1918–1919 гг. в Германии и отмены Брестского договора 1918 г. начался отвод оккупационных войск за реку Березина. 10 декабря 1918 г. части Красной Армии вступили в Минск, была восстановлена советская власть. Таким образом, после освобождения территории Беларуси от германских войск в конце декабря 1918 г. большевики отказались от идеи восстановления Западной области (коммуны). Конференция белорусской секции РКП(б), проходившая 21–23 декабря 1918 г. в Москве, вынесла решение о создании на освобожденной территории Западной коммуны Белорусской республики. 1 января 1919 г. в Минске был опубликован Манифест о создании Социалистической Советской Республики Белоруссия (ССРБ), а Западная коммуна и Облисполком-зап были ликвидированы.

Список литературы

1. Административно-территориальное деление Беларуси // Архивы Беларуси [Электронный ресурс]. – Режим доступа : <https://archives.gov.by/home/genealogiya/administrativno-territorialnoe-delenie-belarusi>. – Дата доступа : 30.03.2025.
2. Карелин, Е. Г. Формирование региональной системы Советской власти в ходе революции 1917 года (на материалах западной области России) / Е. Г. Карелин [Электронный ресурс]. – Режим доступа : [bookonline.ru>system/files/node/lecture/10-8](http://bookonline.ru/system/files/node/lecture/10-8). – Дата доступа : 30.03.2025.

Яцков Андрей Андреевич, студент 1 курса ФИТУ БГУИР

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

Подсекция «Великая Отечественная война советского народа»

Председатель:	канд. ист. наук, доцент Николаева Л. В.
Члены жюри:	канд. ист. наук, доцент Литвиновская Ю. И. канд. ист. наук, доцент Куракевич Н. И. канд. ист. наук, доцент Сугако Н. А.
Секретарь	преподаватель Мякинская А. В.

ПОСТ №1 У ВЕЧНОГО ОГНЯ БРЕСТСКОЙ КРЕПОСТИ-ГЕРОЙ

Формирование мемориального комплекса "Брестская крепость-герой" стало важнейшим этапом в процессе увековечивания памяти о подвиге защитников цитадели в июне 1941 года. Торжественное открытие комплекса состоялось 18 сентября 1971 г. Особую символическую нагрузку в мемориальном ансамбле несёт Вечный огонь, зажжённый 25 сентября 1971 г. Примечательно, что священное пламя было доставлено с Марсова поля в Ленинграде (ныне Санкт-Петербург), что подчёркивало преемственность воинской славы между различными этапами отечественной истории. Этот акт имел глубокое идеологическое значение в контексте советской мемориальной культуры периода 1960-1970-х гг. Для молодёжи г. Бреста сохранение памяти о защитниках Брестской крепости является ключевым элементом патриотического воспитания. Одним из ярких символов этой работы стал Пост Памяти №1 у Вечного огня мемориального комплекса «Брестская крепость-герой». Эта традиция берет свое начало в 1961 г., когда пионеры впервые встали на почетную вахту у памятного камня будущего монумента [1]. В целях сохранения Памяти защитников Отечества и военно-патриотического воспитания молодого поколения города 4 марта 1972 г. по инициативе Городского Комитета ВЛКСМ и отдела народного образования Брестского городского исполнительного комитета был образован комсомольско-пионерский Пост №1 у Вечного огня Брестской крепости-герой [2]. Спустя двенадцать лет был открыт музей Поста №1 «Мальчишки бессмертного Бреста» [3]. Переход к молодежному караулу стал поворотным моментом, сформировавшим современный облик Поста №1. Традиция, изначально создававшаяся по военному образцу, постепенно приобрела особый воспитательный характер, превратившись в действенный инструмент формирования исторического сознания подрастающего поколения. Именно тогда сложилась та особая атмосфера торжественности и сопричастности к великому прошлому, которая отличает Пост №1 сегодня. Этот период стал временем становления уникальной системы подготовки юных часовых, где строгая дисциплина сочетается с глубоким изучением истории. Сформировавшийся в 1980-е гг. подход к отбору и обучению караульных сохранил свою актуальность и в наши дни, доказав свою эффективность как инструмент патриотического воспитания. Особенно важно, что именно тогда был выработан тот особый церемониал смены караула, который, претерпев лишь незначительные изменения, продолжает производить неизгладимое впечатление на посетителей мемориала. Таким образом, 1980-е гг. можно по праву считать временем окончательного оформления Поста №1 в том виде, в котором мы знаем его сегодня. 27 января 2005 г. на-

шему Посту Памяти в Бресте было вручено Знамя Молодежного Георгиевского Союза Российской Федерации за патриотическое воспитание молодежи и в честь 60-летия Победы в Великой Отечественной войне 1941-1945 гг. На территории бывшего СССР было организовано 167 комсомольско-пионерских Постов №1. Первым был открыт Пост №1 города-героя Волгограда. Пост Памяти Брестской крепости был образован четвертым. После распада Советского Союза в 1991 г. многие Посты Памяти прекратили свое существование, отдельные спустя годы возобновили свою деятельность. Брестский Пост, один из немногих, не прекращал своей деятельности ни на один день. На сегодняшний день постов осталось около 50. В нашей республике 3 постоянно действующих Поста Памяти: в городе-герое Минске, в Могилеве и в Брестской крепости-герой. В Бресте сложилась уникальная практика, когда каждая школа формирует собственный мемориальный отряд, получающий право поочередно нести недельную вахту у Вечного огня. Всего Вахту Памяти несут 43 мемориальных отряда от учреждений образования города Бреста. В год школу мужества проходят около 1400 учащихся. Мемориальный отряд состоит из 33 человек, в состав которого входит Почетный караул (25 человек): начальник Почетного караула, помощник начальника Почетного караула, комиссар Почетного караула, 2 разводящих, 4 смены (по 2 часовых и 2 подчаска в каждой смене), 2 экскурсовода и 2 дневальных и запасная смена (8 человек), которая привлекается к Вахте по необходимости. Каждый Почетный караул несет Вахту Памяти раз в год, семь дней в неделю (с четверга по среду). Время несения Вахты Памяти у Вечного огня каждой сменой 15-20 минут, в зависимости от погодных условий. Процесс подготовки юнармейцев к несению вахты у Вечного огня представляет собой тщательно организованную систему. Будущие часовые проходят интенсивный курс теоретической подготовки, глубоко изучая исторические материалы об обороне Брестской крепости, знакомясь с архивными документами и символикой мемориального комплекса. Параллельно осуществляется физическая подготовка – специальные тренировки на выносливость, отработка строевых приемов и церемониальных движений. Кульминацией становятся экзаменационные испытания, включающие проверку исторических знаний, демонстрацию строевой подготовки и оценку физической готовности. Эта строгая система отбора гарантирует, что право нести вахту получают действительно достойные представители молодого поколения, что подчеркивает особый статус Поста №1. Несение Вахты Памяти становится для школьников серьезным испытанием, в ходе которого оцениваются все аспекты

их подготовки и службы. Особое внимание уделяется качеству несения вахты, строевой подготовке, дисциплине, состоянию караульного помещения, а также выполнению исследовательских работ. Такая комплексная система оценки превращает несение Вахты в настоящее соревнование между школьными коллективами, где каждый стремится проявить себя с лучшей стороны. Церемония смены караула – это тщательно отрежиссированный ритуал, где воинский устав органично переплетается с символическими жестами памяти. Каждое движение часовых отрабатывается до автоматизма, особое значение придается абсолютной синхронности, точному соблюдению временных интервалов и особой пластике перемещений. По итогам несения Вахты Памяти в году проводят торжественное мероприятие, на котором подводятся итоги и определяются лучшие. Эта традиция награждения наиболее отличившихся отрядов стала важным стимулом для юных участников, для которых право представлять свою школу на таком ответственном посту является огромной честью. Как отмечают сами юнармейцы, подобный опыт требует полной самоотдачи, но дает бесценные уроки ответственности и патриотизма. Современный этап существования Поста №1 демонстрирует удивительную способность традиции адаптироваться к новым реалиям. Если в советский период она имела выраженный идеологический характер, то сегодня акцент сместился на воспитание

гражданской ответственности, формирование исторического сознания, развитие лидерских качеств и организацию межпоколенческого диалога. Анализ современных регламентов показывает, что традиция продолжает развиваться, органично интегрируя новые образовательные технологии при бережном сохранении базовых ритуальных практик. Пост №1 у Вечного огня Брестской крепости-героя – это не просто церемониал, а сложный социокультурный феномен, в котором переплетаются воинский ритуал, педагогическая практика и коллективная память. Его устойчивость в условиях меняющихся идеологических парадигм свидетельствует о глубокой укоренённости традиции в общественном сознании [5].

Список литературы

1. Суворов А. М. Брестская крепость. Война и мир. / А. М. Суворов. – Брест : Полиграфика, 2010. – 255 с.
2. Национальный мемориальный комплекс «Брестская крепость-герой» : [Электронный ресурс]. – Режим доступа: <https://brest-fortress.by/novosti/614-50-let-posturamyati-1>. – Дата доступа: 05.04.2025.
3. Лесин А. А. Музеефикация фортов Брестской крепости: достижения в коммеморации отечественной военной истории // Культура. Наука. Творчество = Культура. Наука. Творчасць = Culture. Science. Arts. – 2022. – С. 163-167.
4. Попов А. Н. Мемориальная культура и историческая память: традиции и трансформации. – М.: Изд-во РГГУ, 2018. – С. 145.

Арсенович П.А., студент 1 курса ФИТУ БГУИР.

Научный руководитель: Сугако Наталья Анатольевна, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

МОЙ ПРАДЕД – ГЕРОЙ!

Давно закончилась Великая Отечественная война, стремительно и неуклонно летит время. Сменяются поколения. Но мы помним тех, кто не жалел жизни ради нас, делал всё, чтобы не было войны, чтобы над нашими головами светило яркое солнце и всегда было чистое небо. День Победы – это день победы наших воинов, это – день памяти о тех днях, когда наш народ героически сражался с фашистскими оккупантами за независимость родной страны. Нет семьи, которой не коснулась бы война. Всенародное горе не обошло стороной моих родных и близких. В кругу нашей семьи этот день отмечается обязательно. Своей героической жизнью прославился мой прадед. Жук Василий Александрович родился в д. Горно Плещеницкого района Минской области в 1908 г. [1]. Работал председателем колхоза. Вместе с женой Гунич Меланией Ильиничной воспитывал пятерых детей: Регину, Николая, Марию, Татьяну и Нину. Он жил, как миллионы молодых людей того времени. Работа была прервана в связи с началом Великой Отечественной войны. В первые дни войны добровольцем вступил в партизанский отряд в составе бригады имени Михаила Илларионовича Кутузова, которая вела свою деятельность в Кричевском районе Минской области [2]. В августе 1944 г., после освобождения Беларуси, отряд присоединился к войскам Красной Армии. Василий Александрович был призван в регулярную армию и продолжил службу на 2-м Белорусском фронте, который находился под командованием Георгия Федоровича Захарова, в 3 батальоне 79 стрелкового полка 321 стрелковой дивизии в звании младшего сержанта в должности пулеметчика. В бою 2 февраля 1945 г. в районе населенного пункта Нагатау (сегодня территория Германии) из своего ручного пулемета уничтожил 18 солдат противника [3]. За что Указом Президиума Верховного Совета СССР №09-М от 15 февраля 1945 г. был награжден медалью «За отвагу». В конце февраля 1945 г.

Жук Василий Александрович был убит и похоронен в братской могиле в населенном пункте Фрихтхорст, в Восточной Пруссии [4]. По воспоминаниям моей прабабушки Марии Васильевны семья получила извещение («похоронку»), которое стало полной неожиданностью для всей семьи: «Ваш муж, м/сержант Жук Василий Александрович, уроженец Минской области дер. Горно в бою за Социалистическую Родину, верный воинской присяге, проявив героизм и мужество, был убит 02.02.1945 г. Похоронен в населенном пункте Фрихтхорст, Восточная Пруссия...». Спустя несколько дней подавленный горем умер отец Василия Александровича. А через месяц после полученной «похоронки» почтальон принес семье посылку с фронта, в котором также находилось письмо моего прадеда. В нем он писал: «...Мне присвоили звание младшего сержанта. И я скоро вернусь домой. Мы лихо бьем врага, до Победы осталось немного...» Сегодня в деревне Задорье Минской области на мемориальной доске воинов Великой Отечественной войны указана фамилия моего прадеда как героя Победы. Помнить и гордиться величественным подвигом советского народа, мужеством и героизмом солдат и офицеров, женщин и детей – всех, кто, не щадя себя, сражался и трудился, приближая ВЕЛИКУЮ ПОБЕДУ, – долг и обязанность нынешних и будущих поколений! Пока есть жизнь на земле, должна жить и память. Память о тех, кто принёс нам освобождение. Память о подвигах, во имя нашей мирной жизни...

Список литературы

1. Подвиг народа [Электронный ресурс]. – Режим доступа : <https://podvignaroda.ru/>. – Дата доступа : 22.02.2025
2. Партизаны Беларуси [Электронный ресурс]. – Режим доступа : <https://partizany.by/>. – Дата доступа : 22.02.2025.
3. Память народа [Электронный ресурс]. – Режим доступа: <https://pamyat-naroda.ru/>. – Дата доступа : 22.02.2025.

Вильтовская Анастасия Дмитриевна, студентка 1 курса ФИТУ БГУИР

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

НИКОЛАЙ ЕГОРОВИЧ БОРАНЕНКОВ: ВОЕННЫЙ ПУТЬ ПИСАТЕЛЯ

Николай Егорович Бораненков, родившийся 27 ноября 1918 г. в селе Липово Дятьковского района Брянской области, оставил значительный след как писатель и военнослужащий. Он получил образование в лесохимическом техникуме и начал свою трудовую жизнь в колхозе. С 1937 по 1939 г. работал литературным сотрудником в районной газете «Фокинский рабочий». В 1939 г. Николай был призван в Красную Армию. Вскоре началась Вторая мировая война, а затем и Великая Отечественная война, и его жизнь приняла драматический поворот. В то время, когда он исполнял воинский долг, его родители были убиты, а старшая сестра Мария погибла за связь с партизанами. Его родное село было сожжено врагом, что оставило глубокий след в его душе и стало основой для творчества. Н.Е. Бораненков воевал на различных фронтах – Северо-западном, а также 1 и 2 Украинском фронтах. В боях за Старую Руссу он проявил мужество, заменив погибшего командира роты и ведя своих солдат в атаку, несмотря на полученное ранение. Был награжден орденами Красного Знамени, Отечественной войны II степени, Красной Звезды и другими наградами. После окончания войны Николай Егорович уехал в Москву, где окончил Военно-политическую академию имени В.И. Ленина, и продолжил свою службу в советской армии. В это время он занимал различные должности, включая разведчика и политрука, участвовал в выпуске боевых листов и корреспонденций для советских солдат. Работал в редакциях окружной газеты и в «Красной звезде», где раскрыл свой творческий потенциал как писатель. Н.Е. Бораненков женился, завел семью и начал создавать литературные произведения. После увольнения в запас в должности полковника, он стал заведующим редакцией художественной литературы издательства «Московский рабочий», а затем перешел на должность редактора в издательство ДОСААФ. Одним из значительных моментов в жизни

Н.Е. Бораненкова стало знакомство с известным военным корреспондентом Михаилом Шолоховым, который, не забывая о своем земляке, рекомендовал его в Союз писателей СССР. Это сотрудничество навсегда оставило след в биографии писателя. Его первая книга «Боевые товарищи» вышла в 1950 г. На протяжении творческой карьеры он издал более десяти сборников рассказов и романов, связанных с темами войны, патриотизма и партизанского движения. Среди его произведений можно выделить романы, такие как «Гроза над Десной», повествующий о партизанах Брянщины, и «Вербы пробуждаются зимой», представляющий собой романтическое изображение русских солдат в последние дни войны. Его книга «Белая калина» рассказывает об истории жизни забытого села Вересовка. Кроме того, он написал сборники юмористических рассказов, таких как «Любовь неугасимая», «Под звездами кулундинскими» и «Синеглазые москвички». Произведения Николая Егоровича наполнены подлинными историями людей, которые жили в его родном районе, среди которых такие личности, как В. В. Шапкин, И. Д. Волков, И. К. Ефименков и И. В. Дымников. Книги Н.Е. Бораненкова отражают не только его личный опыт участника войны, но и коллективный опыт множества людей его поколения, столкнувшихся с вызовами и трагедиями войны. Таким образом, творчество Николая Егоровича Бораненкова представляет собой ценное наследие.

Список литературы

1. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. Н/Д 9145;
2. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. КП 40869 14 1 –4
3. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. НД 14727;
4. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. КП 42603 2.

Дмитриевич Е. В., студентка 1 курса ФИТУ БГУИР.

Научный руководитель: Сугако Наталья Анатольевна, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

БОЕВОЙ ПУТЬ МАРАТА ИВАНОВИЧА КАЗЕЯ В ПЕРИОД ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ

Марат Иванович Казей родился 10 октября 1929 г. в деревне Станьково Дзержинского района Минской области. По словам его двоюродной сестры Нины Ефимовны Воробьёвой – Марат был небольшого роста, тоненький светлоглазый мальчишка с синими глазами. В школе ничем не отличался от своих сверстников. Учился хорошо, был звеньевым в пионерском отряде, всегда серьёзен, подтянут. Он очень много читал и любил петь. Он очень любил свою школу и дорожил её честью. Особенно нравилась ему работа в столярной мастерской. Среди одноклассников у него было много друзей. А ещё его друзьями были солдаты, которые привили ему увлечение к военному делу. Поэтому Марат был организатором всех военных игр, изготовкой деревянного оружия. Вскоре после начала Великой Отечественной войны он потерял мать. Теперь для него единственной целью стала месть немецко-фашистским захватчикам за смерть матери и его друзей. Марат был знаком со многими советскими людьми, направленными из-за линии фронта для создания партизанского отряда. К ним он и ушёл с друзьями из деревни [1]. Двоюродная сестра Марата Казея, Нина Ефимовна Воробьёва писала в своих воспоминаниях, что хорошо помнит, как переживал её двоюродный брат, когда в отряде его определили быть помощником на кухне. Он по 2 раза в день бегал к командиру отряда, просил, сердился. Однако Марата не посылали на серьёзные задания, его берегли, хотя уже хорошо знали о нём по той работе, которую он выполнял как связной. Помнили о том, что ему всего 12 лет. Во время войны Марат подружился с разведчиками отряда. Иногда разведчики брали подростка с собой, когда это было безопасно. И убедились – этот мальчишка стоит каждого из них. Они то и убедили командование партизанского отряда зачислить Марата в разведку. Как он ликовал, когда ему доверили такое ответственное дело.

Здесь, в разведке, он нашел своё место и призвание, часто получал от командования благодарности. Во время войны Марат Иванович Казей проявил себя в марте 1943 г. в бою с немецко-фашистскими захватчиками возле деревни Румок Узденского района Минской области. Окруженный гитлеровцами партизанский отряд попал в очень тяжёлое положение. Все попытки связаться с другими партизанскими отрядами не увенчались успехом. Тогда установить связь с партизанским отрядом имени Фурманова вызвался Марат Казей. Несмотря на невероятные трудности, он вовремя привёл партизанский отряд на помощь окружённому своему отряду. Бой с немцами закончился победой партизан. 11 мая 1944 г. Марат Иванович Казей принял свой последний бой. Гитлеровцы, преследовавшие юного разведчика, хотели взять его живым. Марат отстреливался до последнего патрона. Потом в ход пошли ручные гранаты. Последнюю гранату он взорвал в своих руках, чтобы не сдаться врагу. В этом бою Марат Казей уничтожил больше десятка немецких солдат и офицеров [2]. Указом Президиума Верховного Совета СССР от 30 декабря 1948 г. Марат Иванович Казей был посмертно награждён медалью «За отвагу». Президиум Верховного Совета СССР Указом от 8 мая 1965 г. за особые заслуги, мужество и героизм, проявленные в борьбе против немецко-фашистских захватчиков в годы Великой Отечественной войны (1941–1945) присвоил М.И. Казею звание Героя Советского Союза (посмертно) [3].

Список литературы

1. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. Н/Д 9145.
2. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. НД 14727;
3. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. КП 426032.

Двораковский Д. А., студент 1 курса ФИТУ БГУИР

Харкевич А. Д., студент 1 курса ФИТУ БГУИР

Научный руководитель: Сугако Наталья Анатольевна, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

ТРОСТЕНЕЦ – СИМВОЛ ТРАГЕДИИ В МОЕЙ СЕМЬЕ

Сегодня остается все меньше людей, испытавших на себе тяготы заключения в Тростенецком лагере. У нового поколения память о войне не такая яркая. Оно должно знать правду о героических и трагических событиях минувшей войны. Это вечный долг человечества перед памятью миллионов жертв нацизма. Однажды я от своего дедушки услышала страшную историю, произошедшую с моей прабабушкой Софьей Ивановной Капустик, которая в годы Великой Отечественной войны стала жертвой лагеря смерти Тростенец. Он был крупнейшим местом массового уничтожения людей на территории СССР и БССР в годы немецко-фашистской оккупации. Название «Тростенец» объединяет несколько мест массового уничтожения населения: урочище Благовщина – место массовых расстрелов; фактический лагерь – у села Малый Тростенец, в 10 км от Минска по Могилевскому шоссе; урочище Шашковка – место массового сожжения людей [1, с. 3]. Среди жертв Тростенецкого концлагеря были мирные жители Минска и его окрестностей, многочисленные подпольщики и партизаны, тысячи советских военнопленных. Также анализ источников дает основание сделать вывод, что в Тростенецком лагере смерти погибло не менее 80 тыс. иностранных, а также не менее 60 тыс. белорусских евреев. Это позволяет говорить о том, что система Тростенецкого лагеря стала одной из основных в политике Холокоста в Европе. По последним данным общее количество жертв лагеря достигало 546 тыс. человек. Моя прабабушка, Капустик Софья Ивановна, родилась 17 ноября 1924 г. в деревне Сеница. В 1941 г., в возрасте 17 лет, окончила 10 классов. После оккупации Минска Софья Ивановна вступила в минское подполье, где была связной, принимала участие в боях с фашистами. В 1943 г. соседка рассказала немцам о деятельности моей прабабушки и ее арестовали гестаповцы. После мучительных пыток информацию Со-

фья Ивановна не выдала, ее отправили в концлагерь Тростенец. По ее рассказам, печи крематория не работали, поэтому группу отправили в деревню Тростенец на расстрел. Группу согнали в одно место и начали огонь. Мою прабабушку ранили. От болевого шока она потеряла сознание. Вечером на место расстрела пришли местные жители, чтобы закапывать тела, и обнаружили мою прабабушку под грудой трупов. Пожилая пара из деревни забрала ее к себе и лечила на протяжении двух месяцев. Когда моя прабабушка поправилась, она хотела уйти в партизанский отряд, но не успела. Осенью 1943 г. немцы окружили деревню и многих жителей забрали в Германию на принудительные работы. Полтора года моя прабабушка находилась на территории Германии в качестве рабыни. И только после Победы в 1945 г. Софья Ивановна вернулась на Родину. После издевательств фашистов возвратилась инвалидом. Она была награждена Орденом Отечественной Войны II степени и медалями. В мирное время моя прабабушка закончила Минский педагогический институт. Затем она на протяжении 40 лет работала учителем белорусского языка и литературы в деревне Луговая Слобода (Минский район). После выхода на пенсию она работала библиотекарем в той же школе. Лагерь смерти Тростенец оставил страшный след в моей семье. Память об этих событиях будет храниться всегда в наших сердцах. Именно потому задача молодого поколения – помнить об ужасах геноцида в годы войны и не допустить их повторения, сохраняя мир. Подтверждение исторической правды о прошедшей войне выступает неотъемлемой частью общих усилий народов в борьбе за мир, за предотвращение новой войны

Список литературы

1. Лагерь смерти Тростенец : документы и материалы / сост. : В. И. Адамушко [и др.] ; под ред. Г. Д. Кнатько; редкол. : В. И. Адамушко [и др.]. – Минск : НАРБ, 2003. – 260, [1] с., [14] л. ил., цв. ил., портр., факс..

Ефимчик Ксения Павловна, студентка 1 курса ФИТУ БГУИР

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

ИСПОЛЬЗОВАНИЕ ТРУДА УЗНИКОВ В КОНЦЛАГЕРЯХ ГЕРМАНИИ

Современные события в мире свидетельствуют о том, что Победа советского народа над гитлеровской Германией ознаменовала собой начало нового этапа в мировой истории. И чем меньше остаётся живых свидетелей прошлого, тем большей должна быть наша забота о сохранении фактов той войны. К сожалению, в мире есть силы, готовые переписать страницы истории, обеляя нацизм. Концлагерь фашистской Германии – явление уникальное и по своим масштабам, и по своей жестокости. Необходимая для подавления инакомыслия в тоталитарном государстве Третьего Рейха страшная система была признана мировой общественностью преступлением против человечества. Исследователи насчитывают более 60 видов лагерей Третьего Рейха: лагеря для военнопленных (Дулаги, Шталоги, Офлаг), собственно концентрационные лагеря (Дахау, Бухенвальд, Маутхаузен) и лагеря для массового уничтожения евреев в рамках «Окончательного решения еврейского вопроса». С 1933 по 1945 гг. в концентрационных лагерях содержалось более 1,5 млн. заключенных, 1,2 млн. из них были убиты [1]. Лагеря были разбросаны по всей территории Третьего Рейха, имели похожую структуру управления и организации. Однако каждый концлагерь имел свою специфику условий, национального состава заключенных, рабочей занятости узников. Так как труд узников был важным производственным ресурсом, у каждого концлагеря было своё функциональное предназначение (осушение болот, добыча камня, производство). Использование рабского труда заключённых было принято многими производственными концернами (например, Siemens). Тысячи несчастных узников, недоедавших, недосыпавших, подвергавшихся по малейшему поводу физическим наказаниям, круглосуточно работали на Германию. За малейшее нарушение режима часовые расстреливали пленных, тела которых увозили в крематорий. Строительство подземных заводов началось с определенной концепции, первоначально сугубо умозрительной, поскольку в индустриальной традиции Германии не было тенденций размещать крупное и сложное производство под землей. Подземелья были прерогативой горнодобывающей промышленности: разнообразных шахт и рудников, добывающих уголь, железную руду. Понимание необходимости зарываться под землю, скрываясь от бомбардировок противника, стала очевидным еще в 1920-е гг., когда шло осмысление опыта Первой мировой войны и строились планы войны будущего. Именно доктрина Джулио Дуэ была положена в основу зарождения идеи Третьего Рейха [2]. Лагерный комплекс Бухенвальд являлся уникальным, первоначально специализирующимся на уничтожении узников через непосильный труд в каменоломнях, а по-

том использовался труд на производстве оружия и строительства железной дороги. К концу войны насчитывалось 136 таких лагерей. Самым известным из них был Дора, где в подземных тоннелях производили ракеты «Фау-2», оружие возмездия Третьего Рейха. Лагерь Дора был создан в 1943 г. в 5 км. от города Нордхаузена в Тюрингии. Он существовал как подразделение лагеря Бухенвальд. Секретностью стройки было обусловлено и отношение к заключенным. Потенциально узник, привлекавшийся к работам на объектах, был носителем секретной информации. А значит, любой ценой оставить его внутри объекта – было одной из обязанностей СС. Многие историки считают, что львиная доля привлеченных на объект заключенных погибла от голода и непосильного труда. Основной задачей жертв становилось строительство военных предприятий. Однако рытьем тоннелей и другой тяжелой работой, от которой заключенные чаще всего умирали через несколько недель, дело не ограничивалось. Производство ФАУ-2 требовало наличия сборочной линии и определенной скорости работы, которой было бы достаточно, чтобы дать Рейху столько ракет, сколько требовал фюрер. Это требование во многом определило эффективность всей программы по созданию «оружия возмездия» и его реальных перспектив. Несмотря на то, что боевая часть ракет: приборы, двигатели, и многое другое, на завершающем этапе монтажа изделия настраивались немецкими специалистами-ракетчиками, но массовую сборку корпусов, элементов управления и других жизненно важных систем доверяли заключенным. Узники, привлекаемые к сборке ракет, хорошо понимали, что даже относительно небольшой дефект может вызвать взрыв ракеты не только в небе, но и на стартовом столе. Поэтому они иногда пользовались случаем, чтобы повредить ключевые элементы управления. На строительство авиационных заводов пленных свозили вплоть до конца войны. Многие из них не дожили до освобождения. Фотографии умерших и живых потрясли весь мир. Изможденные и доведенные до крайнего истощения люди не могли стоять на ногах. В результате пыток, медицинских экспериментов и истощения в Бухенвальде погибло 56 тыс. человек. До момента освобождения дожила 21 тыс. человек. В конце войны это был самый крупный лагерь смерти. В один из филиалов Бухенвальда попал мой земляк Квятчинский Борис Андреевич. О вероломном нападении немецких войск на Советский Союз семья Квятчинских узнала из объявления по радио. Во время Великой Отечественной войны город Пинск и его окрестности находились под оккупацией немецких войск. Когда Квятчинскому Борису было всего 16 лет, на площади г. Пинска он попал в облаву. 17 мая

1944 г. его насильно угнали в Германию, где он попал в концентрационный лагерь «Бибра» в окрестностях г. Кала в Тюрингии. Сотни военнопленных участвовали в масштабном проекте по строительству авиазавода под названием «Ласх», в переводе с немецкого «лосось». Здесь по планам архитекторов Третьего Рейха должен был быть завод по производству новейшей модели истребителей, чтобы отвоевать небо. Пленные жили рядом с заводом, работали от зари до зари, так что не видели дневного света. Из воспоминаний Квятчинского Б.А.: «Кроме нас, граждан СССР, там были бельгийцы, итальянцы, французы. Лагерь, где нас держали, был большой, по площади обнесён колючей проволокой. Внутри лагеря стояли пулеметные вышки. Охраны было мало. Спали в палатках на земле. Костры разводить запрещали, так как опасались авианалётов. Кормили настолько плохо, что единственной постоянной мыслью была мысль о еде. Кормили брюквой, кониной и даже лягушками, и все это – в мизерных количествах». Борис Андреевич вспоминал, как американские самолеты бомбили немецкие склады и аэродромы. 3 мая 1945 г. пленные лагеря Бибра были освобождены американскими войсками. После освобождения русских пленных поселили в помещении бывшего завода. Американцы накормили узников и переделали. Борис Андреевич был перевезён в

восточную зону Германии, где некоторое время работал на фермерском хозяйстве. После окончания войны фермер купил ему билет на поезд и отправил в Беларусь. После возвращения на родину Борис Андреевич стал капитаном речного теплохода БТ-0250, где и проработал до пенсии, до 1988 г. По словам родственников семьи, сам Борис Андреевич Квятчинский не любил вспоминать военные годы, так как это причиняло ему боль. По данным Белорусского государственного музея истории Великой Отечественной войны, за время нацистской оккупации на территории Беларуси погибли 1 миллион 547 тысяч гражданских лиц и 810 тысяч военнопленных, большинство в различных местах уничтожения и принудительного содержания. 11 апреля считается международным днем освобождения узников фашистских концлагерей. Это дань памяти всем тем, кто вытерпел ужасы плена и выжил.

Список литературы

1. Дороги смерти: концлагеря фашистской Германии; информационно-библиографический сборник; сост. Тараканова Г. С. – Абакан, 2010 . – 64 с.
2. Вотье А. М. Военная доктрина генерала Дуэ. – Санкт-Петербург, 2003. – 240 с.
3. Семейные архивы семьи Квятчинского Б. А.
4. Зональный государственный архив в г. Пинске – Ф. 118. Оп. 1. Д. 4. Л. 63

Ивашевич Д.С., студент 1 курса ФИТУ БГУИР.

Научный руководитель: Сугако Наталья Анатольевна, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

ФЕНОМЕН СТАТИСТИКИ НЕМЕЦКИХ И СОВЕТСКИХ ЛЕТЧИКОВ-АСОВ ПЕРИОДА ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ

Несмотря на то, что за прошедшее время многое изменилось в мире, по-прежнему не ослабевает интерес к событиям минувшей войны. Особый интерес представляет феномен статистики немецких и советских летчиков-асов периода Великой Отечественной войны. Несмотря на наличие целого ряда работ посвященных действиям авиации, этот вопрос мало рассматривался. Можно выделить две точки зрения на данную проблему: отрицание большого количества воздушных побед немецких асов [3] и попытка их объяснить [1]. Рассмотрим подробнее статистику: 23-летний майор Эрих Хартманн претендовал на 352 сбитых самолета, в том числе 348 советских и четыре американских. Его коллеги по 52-й истребительной эскадре Люфтваффе Герхард Баркхорн и Гюнтер Ралль заявили о 301 и 275 сбитых самолетах соответственно. Эти цифры резко контрастируют с результатами лучших советских пилотов-истребителей: 62 победами И.Н. Кожедуба, 59 – А.И. Покрышкина, по 57 – Н.Д. Гулаева и Г.А. Речкалова. Более подробная информация об асах Люфтваффе еще более внушительная. Известно, что асами в терминологии союзников (то есть сбившими 5 и более самолетов противника) у немцев было более 3000 пилотов. Э. Хартманн и Г. Баркхорн с более чем 300 побед были лишь вершиной айсберга. Еще 13 летчиков-истребителей Люфтваффе одержали от 200 до 275 побед, 92 – от 100 до 200, 360 – от 40 до 100. При этом замечен парадокс: как правило, выше индивидуальные счета пилотов у стороны, которая проиграла воздушную войну. Следует отметить, что число заявленных сбитых самолетов не является сколь-нибудь объективным показателем при сравнении результатов военно-воздушных сил (далее – ВВС) СССР и Германии. Добиться трехзначных счетов летчиков-асов можно было, сознательно выбрав ведение воздушной войны при численном превосходстве противника и постоянных рокировках авиационных частей и соединений с пассивных участков фронта в пекло боев. Но этот подход – оружие обоюдоострое и зачастую приводил к проигрышу воздушной войны. Причину разницы в счетах пилотов можно объяснить различными факторами: 1) Главной причиной было интенсивное использование ВВС Германией. Выполняя шесть вылетов в день, при постоянном перемещении вдоль линии фронта немецкие пилоты могли существенно увеличивать свою статистику. Советские пилоты, выполняя по одному вылету в день и, оставаясь все время на одном и том же участке фронта, естественно, имели намного меньшие шансы встретить и сбить противника [1; 4]. 2) Следует учитывать эффект масштаба. Число сбитых являлось прямо пропорциональным числу це-

лей в воздухе. Немецкая авиация на протяжении всего периода Великой Отечественной войны была вынуждена воевать на два фронта. В результате вероятность немецкого аса встретить в небе противника и увеличить свой счет побед была намного больше, чем у советского пилота. 3) Важным фактором также было общее превосходство немцев в технике на протяжении значительной части Великой Отечественной войны. Немецкие асы с самого начала войны имели в своем распоряжении прекрасный истребитель Мессершмитт Bf 109, на котором летала большая часть асов [5; 7]. В 1941 г. появился также хороший самолет Focke-Wulf 190. При этом Мессершмитт Bf 109 оставался основным истребителем Германии в годы Второй мировой войны. Аналогичных истребителей, за исключением британского Supermarine Spitfire, у СССР и союзников просто не было. Хорошие советские истребители Як 9 и Ла 5 появились в 1942 г. и к 1943 г. были доработаны. Ла 7 и Як 3 начали использоваться только с весны 1944 г. [2; 5] Также ремонтпригодность и прочность советских самолетов была ниже из-за использования древесины в конструкции. 4) Еще одной причиной большого количества побед немецких асов стали различные методы ведения воздушной войны. Пилоты Люфтваффе имели возможность летать в так называемом режиме «свободной охоты». Это позволяло им быстрее находить противника, неожиданно атаковать его и при необходимости скрыться. В то же время советские истребители были «привязаны» к наземным войскам и главной их задачей было сопровождение штурмовиков и бомбардировщиков [1]. 5) Значение имел опыт и уровень подготовки пилотов. Средний объем часов налета немецких пилотов в начале Великой Отечественной войны был существенно выше. Этому способствовал опыт ведения боев на западном фронте в компаниях 1939–1940-х гг. Также была существенная разница между немецкими и советскими пилотами по окончании летных школ. Однако к середине войны ситуация начала меняться. Люфтваффе не успевал восполнять свои потери и средний уровень подготовки летного состава падал. Кроме того была зависимость от летчиков асов, потеря которых приводила к ощутимому снижению боеспособности остальных пилотов эскадры. Советское командование сделало правильные выводы и сумело организовать обучение пилотов в куда большем количестве, что позволяло избежать зависимости от результатов конкретных личностей, и в дальнейшем улучшить качество и методику обучения летного состава. Немецким летчикам просто приходилось больше летать, а потому и получалось больше сбивать. С этих позиций становится вполне очевидным, поче-

му наибольшие успехи германской авиации приходятся на 1943–1945 гг. В этот период летная нагрузка немецких летчиков резко возросла. Это произошло потому, что с 1943 г. советская авиация по своей численности стала резко превосходить германскую. Активность советских ВВС в результате этого существенно увеличилась. Естественно, что пилотам более малочисленных Люфтваффе для противодействия советской авиации приходилось чаще подниматься в воздух [3]. 6) Большое количество побед немецких асов на восточном фронте и более скромные показатели на западном направлении объясняются спецификой боевых действий. На восточном фронте немецкие летчики воевали практически непрерывно. В зависимости от обстановки их тасовали между разными участками фронта сообразно с проводившимися оборонительными или наступательными операциями. На западном фронте немецким пилотам приходилось отражать рейды тяжелых бомбардировщиков (которых было тяжело сбить) в сопровождении истребителей, как правило, насчитывающие несколько сотен боевых машин [6]. В заключении можно сделать вывод о различных способах применения истребительной авиации в годы Великой Отечественной войны. Специфика применения Германией авиации позволила немецким асам одержать большое количество побед. Однако этот путь был вынужденным и итогом его стал триумф советской авиации в небе над Восточным фронтом.

Список литературы

1. Исаев, А. В. Десять мифов Второй мировой / А. В. Исаев [Электронный ресурс] // Милитера – Военная литература. – Режим доступа : http://militera.lib.ru/research/isaev_av2/08.html. – Дата доступа : 17.03.2025.
2. Истребители Второй Мировой войны: лучшие из лучших. Взгляд инженера [Электронный ресурс] // Военное обозрение. – Режим доступа : <https://topwar.ru/7742-istrebiteli-vtoroy-mirovoy-voyny-luchshie-iz-luchshih-vzglyad-inzhenera.html>. – Дата доступа : 17.03.2025.
3. Мухин Ю. И. Асы и пропаганда. Дутые победы Люфтваффе / Ю. И. Мухин. – Москва : «Эксмо», 2004. – 480 с.
4. Смирнов В. Асы. Загадка успеха воздушных асов / В. Смирнов [Электронный ресурс] // Большая авиационная энциклопедия. – <http://www.airwar.ru/other/article/zuva.html>. – Дата доступа : 17.03.2025.
5. Соболев, Д. А. История самолётов 1919 – 1945 / Д. А. Соболев [Электронный ресурс] // Электронная библиотека. – Режим доступа : <https://www.litmir.me/br/?b=175072p=1>. – Дата доступа : 14.09.2020.
6. Советская авиация в Великой Отечественной войне 1941–1945 гг. в цифрах [Электронный ресурс] // Милитера – Военная литература. – Режим доступа : <http://militera.lib.ru/h/sovaviation/index.html>. – Дата доступа : 17.03.2025.
7. Сравнение истребителей Второй мировой войны [Электронный ресурс] // Авиация Второй мировой. – Режим доступа : <https://airpages.ru/ru/fighters.shtml>. – Дата доступа : 16.09.2020.

Концевич Кирилл Петрович, студент 1 курса ФИТУ БГУИР.

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

СУДЬБА МОЕГО ПРАДЕДА ВО ВРЕМЯ ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ

Мой прадед, Спичаков Павел Софонович, родился 22 ноября 1917 г. в деревне Золотва Черноборского сельского совета Быховского района Могилевской области. В 1940 г. он женился на моей прабабушке Елизавете. За первые четыре дня войны призывные пункты Могилева направили в Красную Армию около 25 тысяч человек. С 24 июня по 3 июля 1941 г. в Могилеве находился штаб Западного фронта. За семь дней вокруг города были созданы две линии оборонительных рубежей. Формировалось народное ополчение, в которое влилось около 12 тысяч жителей области. В июле 1941 г. мой прадед Павел ушел в партизаны. Они устраивали засады, минировали дороги, обрывали связь. В июле 1943 г. в Быховском районе в соответствии с постановлением Могилевского подпольного обкома КП(б)Б и приказом Могилевской (областной) военно-оперативной группы на базе 810, 425 и 225-го отрядов 11-й Быховской бригады и 152-го отдельного отряда была создана Быховская военно-оперативная группа (ВОГ). При формировании Быховской ВОГ управление 11-й Быховской бригады было упразднено, а ее отряды включены в состав ВОГ самостоятельными боевыми единицами. В октябре 1943 г. отряды 810, 425, 152-й были развернуты в партизанские полки под тем же наименованием. ВОГ действовала в Быховском, Могилевском, Кировском районах Могилевской и Рогачевском районе Гомельской областей. 24 февраля 1944 г. Быховская ВОГ соединилась с частями Красной Армии в составе 810, 425, 152-го полков и 225, 435, 820, 830-го отдельных отрядов общей численностью 3554 партизан. 19 декабря 1943 г. у моего прадеда и прабабушки родился старший сын Николай. Это событие еще больше придало ему сил и отваги для защиты Родины. 14 августа 1944 г. мой прадед был в 18-й армейский запасной стрелковый полк

для прохождения дальнейшей службы. Буквально в первые дни службы он получил ранение в руку, но в госпиталь попал только 31 августа 1944 г. Там он прошел лечение. Правда, осколок остался, врачи его не удалили. По воспоминаниям бабушки, прадед (ее отец) ей рассказывал, что в начале руку хотели полностью ампутировать, но в госпитале был один молодой доктор, который сказал что сможет сохранить руку. И это у него получилось. Прадед был ему очень благодарен. 16 сентября 1944 г. прадеда перевели обратно в 18-й армейский запасной стрелковый полк для дальнейшей службы. А оттуда 27 сентября 1944 г. его перенаправили в 269-ю стрелковую дивизию, в которой он дослужил до Победы. После Победы прадед вернулся в родную деревню к жене и сыну. Восстанавливал дом, работал в колхозе. Осенью 1947 г. у них родился второй сын Леонид, а в марте 1951 г. первая дочь Галина (моя бабушка). В августе 1959 г. родилась вторая дочь Вера. 6 апреля 1985 г. моему прадеду Павлу вручили Орден Отечественной войны I степени. Однако 31 мая 1989 г. он скончался от перитонита. Он успел познакомиться со всеми своими внуками и внучками, кроме моей мамы. После смерти прадеда прабабушка подарила его орден моей маме на память о прадедушке. Когда мама училась в СШ № 11 г. Могилева при школе был и до сих пор существует «Музей обороны города Могилева в июле 1941 г.», основанный 27 июня 1964 г. Мама решила передать орден в этот музей, чтобы все могли видеть награду ее бабушки. Орден по сей день хранится в этом музее.

Список литературы

1. . Память народа [Электронный ресурс]. – Режим доступа : <https://pamyat-naroda.ru/>. – Дата доступа : 31.03.2025.
2. Материалы семейного архива автора.

Пейганович Кирилл Александрович, студент 1 курса ФИТУ БГУИР.

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

ОСВОБОЖДЕНИЕ НОВОГРУДКА В ИЮЛЕ 1944 Г. ПО РАССКАЗАМ ОЧЕВИДЦЕВ

В год 80-летия Победы советского народа в Великой Отечественной войне очень актуален вопрос, какой ценой далась победа. Важен каждый исторический факт, каждое событие Великой Победы. И в первую очередь, мы должны знать, как освобождались от немецко-фашистских захватчиков наши родные места, наша малая Родина. 8 июля 1944 г. – одна из ярких дат, которая вписана в историю нашей малой Родины, г. Новогрудка – день освобождения от немецко-фашистских захватчиков. Из воспоминаний начальника артиллерии 856 стрелкового полка 283 стрелковой дивизии, майора Мельникова Александра Леонтьевича: «После освобождения Корелич командир полка создал новый передовой отряд по преследованию противника. Часть роты посадили на самоходку, вторую на автомашины, отбитые у гитлеровцев, двинулись в направлении Новогрудка. При выходе к д. Рутка, передовой отряд столкнулся с немецким пехотным батальоном. Отряд атаковал, и их вражеские колонны были разгромлены. 8 июля в 14.10. передовой отряд с юга ворвался в Новогрудок, а затем подошли основные силы Красной Армии. Подтянутыми резервами противник пробовал отбить город. Бой принял жестокий характер. Была поставлена задача отбить контратаку противника со стороны Новоельни. В этом бою победило мужество стрелков, артиллеристов, минометчиков и воинов. Перед городом были подбиты два танка и шесть бронетранспортеров, 12 автомашин, четыре орудия. Но и нам этот бой обошелся дорогой ценой. Список потерь пополнился именами 36 однополчан, 7 бойцов пропали без вести» [1, с. 116]. Из воспоминаний Корзуна Георгия Сергеевича: «Фронт стремительно приближался к Новогрудку. Люди говорили друг другу, что Красная Армия уже в Столбцах, потом в Кореличах. На следующий день, проснувшись, мы увидели, что за деревней все горит и появились танки, летели самолеты. Бабушка сказала, что будут осво-

бождать Новогрудок и что на перекрестке дорог из Селицкого леса и за Лагодками идет бой, и красноармейцам помогают партизаны. Говорила мать, что в этом бою было убито 400 фашистов и один партизан». Из воспоминаний Скока Николая Ивановича: «Красная Армия освободила Кореличи, и фашисты понимали, что советские войска будут наступать из Корелич на Новогрудок, через Божовичи. Немцы начали рыть укрепления и в сторону Сенежиц от шоссе хорошо укрепили свои позиции. Возле маяка за д. Волковичи в сторону д. Рутка, немцы также вырыли окопы и построили укрепления. Защищать эти позиции немецкое командование оставило штрафбатальон. 7 июля возле Волкович появилась советская разведка, и местные жители рассказали, где немцы укрепились для обороны. Тогда советское командование изменило тактику. Они разделились на две группы. Одна группа войск пошла в сторону Негневиц на Любчу, вторая в сторону Велевки и Невды. Немцы разгадали маневр Красной Армии окружить их в районе Новогрудка, и спешно отступили на запад. А потом возле кладбища появились несколько советских танков, которые уничтожили два немецких орудия. Остальные немцы забрали с собой, и штрафбатальон без боя отступил». Таким образом, события освобождения г. Новогрудка оставили глубокий след в памяти как воинов, освободивших город, так и мирных жителей, ставших свидетелями этих боев.

Список литературы

1. Мельников, А. Л. Огнем полковых батарей : [856 –й стрелковый полк 283 –й Гомельской стрелковой дивизии] / А. Л. Мельников; [Лит. запись И. М. Ботова]. – Киев : Политиздат Украины, 1984. – 330 с. : 2 л. ил.
2. Памяць. Навагрудскі раён : гісторыка-дакументальныя хронікі гарадоў і раёнаў Беларусі / [рэдкал.: Э. Н. Гнеўка і інш.]. – Мінск : Беларусь, 1996. – 558 с., 1 л. партр.

Перко Андрей Дмитриевич, студент 1 курса ФИТУ БГУИР.

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

ЛЕТОПИСЬ ВОЙНЫ В МОЕЙ СЕМЬЕ

В моей семье бережно хранятся документы, фотографии, личные вещи и дневники, написанные моим прадедушкой Беляшовым Василием Мартемьяновичем. В июне 1941 г. проходили выпускные экзамены у советских учащихся. Моему прадедушке оставалось сдать два экзамена и получить диплом агронома. Это была его заветная мечта. Но ей не суждено было исполниться. Началась война. 29 июня Василий Беляшов получил повестку в райвоенкомат. «Ухожу на фронт, береги сына», – телеграмму такого содержания прадедушка отправил своей жене. В его военных документах значилось, что он «годен, но не обучен». Василий Мартемьянович был зачислен в подвижной полевой госпиталь химиком-красноармейцем. Эшелон с врачами и санитарями был отправлен в Ржев, туда, где происходили самые ожесточенные бои с противником. «По пути следования видели разрушенные, сожженные железнодорожные станции, пожарища. В западном направлении двигались походным строем маршевые роты. А по обочинам дороги колосилась высокая рожь, среди которой, как затейливый узор на ковре, красовались нежные васильки, желтый молочай и темно-сиреневые соцветья осота, порхали бабочки». Вчитываясь в эти слова. Осознавал ли тогда молодой мужчина, что его ждет впереди? Думаю, что нет. Его мысли были заняты семьей. Он волновался, как родные будут без него жить. Читаю строки из письма: «Коз не продавайте, запасите для них больше кормов на зиму. Если гряды в огороде залужели, разбороните их железными граблями и присыпьте навозом – будете зимой с овощами». В. Беляшов был назначен старшим писарем, регистрировал поступающих раненых, ездил на железнодорожный вокзал за ними: «На путях горели цистерны с горючим, рвались боеприпасы, горела сама земля, все было окутано густым черным непроницаемым дымом. Среди этого моря огня и дыма находились вагоны с ранеными. Мы и добровольцы из населения спасали их». Весной 1942 г. Василий был переведен в стрелковый полк. Однажды его часть попала под сильный минометный огонь. Встала задача перейти линию фронта, чтобы попасть к своим. Это было очень трудно сделать, потому что немцы беспорядочно обстреливали лес из автоматов и минометов: «Мне было стыдно перед товарищами за то, что в схватку вступил последним, не сделал ни одного выстрела, но зато первым кинулся к брошенной повозке, набрал продуктов и быстрее всех убежал в лес. Ведь я пошел на войну необученным, а мои спутники по несчастью были кадровыми офицерами, имели военный опыт. Наверное, они понимали мою растерянность и потому не упрекали». Добраться к своим частям не удалось. В результате взрыва вражеской гранаты прадедушка был ранен и попал в плен. 18 июля 1942 г. его отконвоировали в лагерь для

военнопленных на станцию Оленино. С этого дня начался отсчет его жизни как узника фашистского лагеря. Из Оленино военнопленных перевели в г. Ржев, где находился пересыльный пункт. Здесь стали сортировать по национальному признаку. Украинцам и крымским татарам предлагали вступить во власовскую армию, добровольцам обещали немецкое обмундирование и паек. Было несколько человек, которые согласились перейти на сторону врага. По мере наступления Красной Армии пленных все время перемещали с одного места в другое. Так, они находились в Орше, затем были перевезены в Борисов, а через три месяца – в Литву, в город Кальвария. Условия в лагерях были ужасные: «Спали на нарах, ели баланду, постоянно подвергались унижениям». О положении на фронтах узнавали от вновь прибывших военнопленных. 14 марта 1943 г. 1500 офицеров-военнопленных Советской Армии, среди которых был и мой прадедушка, оказались в лагере «Шталаг-317» в Австрии. «Ехали четыре дня, почти на каждой большой остановке из вагонов выносили трупы пленных, умерших от ран, голода и холода, скученности. Горячую пищу в виде жидкой каши и 200 граммов хлеба давали один раз в сутки». Вместе с другими невольниками он работал на бетонировании высокогорной штольни, занимался ремонтом горных дорог, возводил из камня опоры, предохраняющие от оползней, очищал бассейн-отстойник от накопившегося в нем ила. «Это была очень грязная и тяжелая работа. Слой ила достигал 80 и более сантиметров, в вагонетки грузили его лопатами, выкатывали на борт бассейна и вываливали». В этом местечке дедушка пробыл до 10 января 1945 г., затем всех пленных перевезли в лагерь «Голенг». В это время уже происходило освобождение Европы от фашизма. Единственным источником информации были листовки, которые сбрасывались с самолетов. В них «сообщалось, какие немецкие города захватили русские, сколько немецких солдат и офицеров взято в плен, и другие новости». Для заключенных они являлись очень важной составляющей в поддержании духа веры в скорое освобождение. Параллельно фашисты старались сломить волю узников лагеря. Для этого к ним регулярно приходил офицер власовской армии, и рассказывал, что «в рядах РОА миллионы бывших советских солдат, и она в скором времени разгромит Красную Армию, освободит все народы России и других республик от коммунистического гнета. В конце он призвал нас вступить в Русскую освободительную армию». Однако никто не принимал за правду эту агитацию. Применялись угрозы, насилие, но изменников среди узников лагеря не нашлось. Невозможно без слез читать следующие строки дневника: «8 мая 1945 г. было тепло, моросил мелкий дождик. Рано утром мы вдруг обнаружили, что на вышках по углам

лагеря нет немецких часовых, не стояли они и у главного входа в лагерь. В чем дело? И тут узнаем радостную весть: кончилась война, фашистская Германия повержена! – Победа! Победа!!! – кричали обезумевшие от радости узники. Все, кто находился в лагере, бросились друг к другу, обнимались, плакали и говорили, говорили...». Советские узники, находясь в ожидании возвращения домой, приняли решение увековечить память о погибших в застенках лагеря товарищах: «Чтобы не терять время зря, мы на месте погребения наших товарищей воздвигли памятники: один на всех 3500 умерших и отдельно на кладбище, где были расстреляны первые 67 человек. При открытии их все, кто мог ходить, выстроились в колонну и под звуки лагерного духового оркестра прошли возле них, отдавая дань памяти соотечественникам, погибшим в этом фашистском лагере». Только 16 июня 1945 г. начался долгий путь домой. Дедушка знал, все военнопленные проходили очень строгую государственную проверку. Полицаев и «власовцев» отправляли под военный трибунал, а тех, у кого совесть была чиста, демобилизовали по Указу Верховного Совета СССР. Среди демобилизованных значился В. М. Беляшов. 10 декабря 1945 г. В. Беляшов вернулся в город Североуральск, где жила его семья. «Не забыть, как я делал последние километры к дому, как ступил на порог его...». «Подошел к двери, постучался. А сердце,

казалось, вот-вот выпрыгнет из груди, в глазах защипало. Кто-то открыл дверь. Первой с радостными восклицаниями кинулась ко мне жена; затем на моей шее повисло сразу несколько человек. Кто-то целовал, кто-то плакал, а я ничего не слышал, ничего и ничего не понимал...». Менее чем через месяц дедушка был принят на работу заместителем главного бухгалтера Второго Северного рудника. Однако, несмотря на хорошо складывающуюся послевоенную жизнь, была у него одна нереализованная мечта – ему очень хотелось получить агрономическое образование. И в 1956 г. он поступил на третий курс заочного отделения Красноуфимского сельскохозяйственного техникума. Закончил его в 1960 г., получил диплом агронома-полевода и успешно применял полученные знания, а позже собственноручно посадил лиственничную аллею, которая до сих пор украшает Североуральск. Мой прадедушка прожил долгую, успешную жизнь. К сожалению, он всего несколько лет не дожил до моего рождения. Но благодаря своим родителям, я знаю о нем все. В нашей семье есть замечательная традиция: когда мы собираемся все вместе, мы достаем пожелтевшие тетради и с огромной любовью, гордостью, уважением и благодарностью «вчитываемся в историю».

Список литературы

1. Материалы семейного архива автора.

Пинчук Яна Игоревна, студентка 1 курса ФИТУ БГУИР.

Научный руководитель: Николаева Людмила Викторовна, заведующий кафедрой гуманитарных дисциплин БГУИР, кандидат исторических наук, доцент, Mikalayeva@bsuir.by

ГЕНОЦИД НА ТЕРРИТОРИИ ЛЮБАНСКОГО РАЙОНА В ГОДЫ ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ

Работа посвящена одной из самых страшных страниц истории – геноциду мирного населения. Война оставила глубокую рану в душах людей, унесла тысячи жизней, будущее у миллионов. Наше поколение, которому, к счастью, не пришлось лично пережить оккупационный режим, войну и разрушения, не должно забывать историю своего края. События сегодняшних дней в некоторых государствах показывают, что фашизм не только прошлое. Нам очень важно сохранить память о прошлом для будущих поколений. Сегодня, используя современные компьютерные технологии, мы имеем возможность и хотим внести свой вклад в сохранение памяти о тех, кто ценой жизни приближал победу всенародной войны, и в первую очередь, для современного поколения, чтобы и они помнили, и передавали друг другу, так как мало кто задумывается, какое огромное количество жертв среди мирного населения она оставила. Содержание работы направлено на сохранение исторической памяти о событиях Великой Отечественной войны на территории Любанского района и привлечение внимания к теме геноцида через разработку мобильного приложения «INMEMORY». Материалы работы представлены в виде основных направлений злодеяний захватчиков на территории района, представленные в трех главах. За годы оккупации немецко-фашистские захватчики замучили и расстреляли 3725 мирных жителей Любанщины, угнали на каторжные работы в Германию 1803 юношей и девушек. В ходе работы был изучен и систематизирован материал о фактах геноцида мирного населения Любанщины. Политика геноцида на оккупированной территории Беларуси в годы Великой Отечественной войны включала карательные операции и тысячи сожженных деревень, лагеря смерти, гетто и другие места принудительного содержания, показательные казни и устрашающие виселицы. Любанщина не стала исключением. Были изучены и систематизированы материалы о фактах геноцида мирного населения Любанщины, исследованы факты злодеяний нацистов на территории Любанского и Уречского гетто; найдены и систематизированы материалы о сожженных деревнях на территории Любанского района; изучены и систематизированы материалы о проведенных на территории Любанского района карательных операциях по уничтожению мирного населения; записаны воспоминания мирных жителей, переживших эти события; проанализированы архивные материалы музея, статьи газет за разные годы по данной тематике, а также воспоминания свидетелей. Был создан интерфейс мобильного приложения и разработано приложение в среде программирования MIT App Inventor с облачным хранилищем документов и удаленной базой данных, которая содержит

сведения о населенных пунктах, уничтоженных полностью и частично, и фактах геноцида в отношении мирного населения на территории Любанского района в годы Великой Отечественной войны. Результатом работы стала разработка и создание мобильного приложения «INMEMORY», в котором размещена вся найденная и систематизированная информация. Особенностью приложения является тот факт, что оно использует облачную базу данных, реализованную в обычных Google Таблицах, что позволяет продолжать вести работу по пополнению новой информацией и редактированию существующих записей без обновления приложения. Для разработки мобильного приложения выбрана среда облачного визуального программирования MIT App Inventor, так как она позволяет создавать приложения под управлением операционной системы Android на основе базовых знаний методов алгоритмизации и программирования. Разработана схема и интерфейс мобильного приложения, а затем в среде MIT App Inventor написан программный код приложения. В результате проведенной работы: создана интерактивная карта сожженных деревень со списками жертв мирного населения на территории Любанского района. За годы оккупации было сожжено 63 населенных пункта. Один населенный пункт Боянички ранее не значился в списках сожженных деревень, но в ходе изучения архивных документов (Приложение 4, Акт о зверствах немецко-фашистских захватчиков от 10 марта 1944 г), обнаружено, что деревня была сожжена 16 февраля 1944 года с тремя ее жителями; собраны материалы о карательных операциях и размещены на интерактивной карте направления движения немецких частей в ходе проведения карательных операций «Бамберг», «Русалка» и «Марабу»; отмечены на интерактивной карте места нахождения Любанского и Уречского гетто и места уничтожения еврейского населения; создана электронная база данных с фамилиями еврейского населения, уничтоженного на территории Любанского и Уречского гетто. Обнаружено, что в списках погибших в Любанском гетто числится Комиссар Соня 1930 года рождения, которая на самом деле выжила и найдены ее воспоминания; создана электронная база данных, которая содержит сведения о населенных пунктах, уничтоженных полностью и частично, и фактах геноцида в отношении мирного населения. Всего в базе данных содержится информация о 63 сожженных деревнях и 18 деревнях, жители которых были убиты; создана интерактивная карта мест памяти – мемориалов, памятников и мемориальных досок, на которой можно увидеть изображения памятных мест, познакомиться с описанием памятников и историческими сведениями, узнать списки захоронений; создана электронная база данных «По-

иск погибших», где восстановлены 2532 фамилии из 3725 жертв геноцида. Пользуясь базой данных по фамилии можно узнать информацию обо всех людях с такой фамилией, которые захоронены на территории нашего района; собрана информация о судьбе 12 мирных жителей, которым удалось выжить и записаны их воспоминания. Хочется верить, что эта работа заставит каждого задуматься и осознать, что важно хранить память о героическом подвиге прошлого поколения и чтить память погибших.

Список литературы

1. Автоматизированный банк данных «Книга памяти» Республики Беларусь. [Электронный ресурс] / Министерство обороны Республики Беларусь, 2022. – Режим доступа: <https://www.mil.by/base/>. – Дата доступа: 19.08.2024.
2. Архив ГУ «Любанский музей народной славы». – Акт о зверствах фашистско-немецких захватчиков над белорусским народом от 10 марта 1944 года – Ф. 68. Оп 2. Д 8. Л.137.
3. Беларусь сегодня. Так убивали людей. [Электронный ресурс] / Беларусь Сегодня, 1998 –2022. – Режим доступа: <https://www.sb.by/articles/tak-ubivali-lyudey-12.html>. – Дата доступа: 19.08.2024.
4. Белорусские деревни, сожженные в годы Великой Отечественной войны. [Электронный ресурс] / Национальный архив Республики Беларусь. – Режим доступа: <http://db.narb.by/>. – Дата доступа: 19.11.2024.
5. Мое местечко. Проект «Голоса еврейских местечек. Минская область». [Электронный ресурс] / Центр «Мое местечко», 2009 –2020. – Режим доступа: http://shtetle.com/shtetls_minsk/luban/knel.html. – Дата доступа: 19.08.2024.
6. Нацистская политика геноцида и «выжженной земли» в Белоруссии (1941 –1944) / редколл. : В. Е. Лобанок [и др.]. – Минск : Беларусь, 1984. – 271 с.
7. Свидетельствуют палачи : уничтожение евреев на оккупированной территории Беларуси в 1941 –1945 г. : Документы и материалы / сост. В. И. Адамушко [и др.]. – Минск : НАРБ, 2009. – 212 с.
8. Трагедия белорусских деревень, 1941 –1944 : Документы и материалы / сост. Н. В. Кириллова [и др.]; редкол. В. И. Адамушко [и др.]. – М. : Фонд «Историческая память», 2011. – 536 с. : илл.
9. Феранц, В. Р. Памяць: Гіст.-дакум. хроніка Любанск. р-на / рэд.-укл. В.Р. Феранц. – Мн. : Ураджай, 1996. – 438 с., [8] л. іл

Попов К.А. , студент 1 курса ФИТУ БГУИР.

Научный руководитель: Мякинська Анна Владимировна, преподаватель кафедры гуманитарных дисциплин

ГЕНАЦЫД БЕЛАРУСКАГА НАРОДА (ДОКШЫЦКІ РАЁН)

Падзеі Вялікай Айчыннай Вайны адгукаюцца ў сучаснай гісторыі Докшыччыны. Так, да 2022 года было вядома пра тры знішчаныя фашыстамі вёскі, якія не былі адноўлены пасля вайны: Шунеўка, Азарцы, Залатухі. Сёння дакладна вядома аб існаванні ды месцазнаходжанні чацвёртай – Баравыя. Звесткі ведамасцей Бягомльскага сельсавета ад верасня 1924 г. адкрылі зводку насельніцтва і гаспадаркі вёскі ў даваенны час – 22 зямельных участка і 96 жыхароў. У выніку, падчас расследавання крымінальнай справы, узбуджанай Генеральным пракурорам Рэспублікі Беларусь па факце генацыду беларускага народа ў гады Вялікай Айчыннай вайны і пасляваенны перыяд, пракуратура Докшыцкага раёна ўстанавіла: вёска Баравыя была сапраўды знішчана ў гады нямецка-фашысцкай акупацыі [1]. Жудасны лёс Баравых паўтарылі 114 вёсак з агульнай колькасцю двароў 4632, расстралялі, спалілі жывымі, закатавалі 21 590 чалавек. Забойствы мірных жыхароў ажыццяўляліся ў працэсе карных фашысцкіх аперацый. Такім чынам, праз тэрыторыю Докшыцкага раёна прайшлі тры знішчальныя паходы: «Котбус» – з 20 мая па 21 чэрвеня 1943 г., «Веснавое свята» – з 17 красавіка па 5 мая 1944 г., «Баклан» – з 25 мая па 23 чэрвеня 1944 г. [2]. Халакост і сёння застаецца страшнай плямай у гісторыі невялікага горада. Паводле перапісу 1939 г. насельніцтва Докшыц складала 3,6 тыс. чалавек, з якіх 75 % – яўрэі. З-за сумненняў аб жорсткасці фашыстаў, большая частка насельніцтва не пажадала эвакуявацца. Ужо 29 траўня 1942 г. створанае ў версані 1941 г. яўрэйскае гета ў Докшыцах было ліквідавана – знішчаны 2653 яўрэі. У чэрвені 1942 г. ў Докшыцах не засталася ніводнага яўрэя. Халакост ажыццяўляўся на ўсёй тэрыторыі Докшыцкага раёна, але знаходзіліся людзі, гатовыя да дапамогі. Так, 14 чалавек з раёна былі ганараваны звання «Праведнік народаў свету» – званне, якое прысуд-

жаецца Інстытутам Яд Вашэм у Іерусаліме неярўэйскім грамадзянам, якія здзейснілі гераічныя ўчынкі, ратуючы яўрэйў падчас Халакосту. Летам 1941 г. у Докшыцкім раёне было створана Параф’янаўскае гета. Тут за час акупацыі было знішчана каля 600 яўрэйў. 30 траўня 1942 г. з гор. Глыбокае прыехала нямецкая жандармерыя на ст. Параф’янава. Пасля збору яўрэйў у будынку клуба – тамсама на станцыі, іх павялі да ямы, дзе адбыўся масавы расстрэл. Так усё яўрэйскае насельніцтва станцыі Параф’янава было знішчана. У чэрвені 1944 г., на фоне набліжэнні Чырвонай Арміі, гітлераўцы праводзілі масавыя карныя аперацыі супраць мірнага насельніцтва і партызан. У в. Ходараўка, недалёка ад станцыі Параф’янава, быў створаны канцлагер, дзе ўтрымваліся савецкія грамадзяне і італьянскія ваеннапалонныя. 27–28 чэрвеня 1944 г. нямецкія карнікі з «Зондэркаманды-9» правялі масавую расправу над зняволенымі. Людзей расстрэльвалі і скідалі ў ямы. Ахвярамі сталі каля 600 савецкіх грамадзянаў з Ушацкага, Лепельскага, Бягомльскага і іншых раёнаў, а таксама 200 італьянскіх ваеннапалонных [3]. Падзеі Вялікай Айчыннай вайны пакінулі глыбокі след у гісторыі Докшыччыны. Карныя аперацыі фашыстаў і халакост сталі трагічнымі старонкамі, якія нагадваюць пра жорсткасць акупантаў і гераізм людзей, што ратавалі іншых. Гэтыя падзеі заахвочваюць шанаваць памяць ахвяр і захоўваць урокі гісторыі для будучых пакаленняў.

Спісок літаратуры

1. Інтэрнэт –партал рэдакцыі газеты «Родныя вытокі» [Электронны рэсурс]. – Рэжым доступу: <https://www.dokshitsy.by>. – Дата доступу: 22.03.2025.
2. Памяць: Гіст.-дакум. хроніка Докшыцкага р-на. –Мн.: БелЭн, 2004. – С. 274-275.;
3. Інфармацыйны рэсурс «Студапедыя» [Электронны рэсурс]. – Рэжым доступу: <https://studopedia.ru>. – Дата доступу: 22.03.2025.

Пракаповіч В.Ю., студэнт 1 курса ФІТіК БДУІР

Ясінскі П.М., студэнт 1 курса ФІТіК БДУІР

Навуковы кіраўнік: Сугака Наталля Анатольеўна, дацэнт кафедры гуманітарных дысцыплін, кандыдат гістарычных навук, дацэнт

РЕКОНСТРУКЦИЯ ИСТОРИИ СЕМЬИ ЧЕРЕЗ ПОИСКОВЫЕ ОНЛАЙН СИСТЕМЫ

Реконструкция истории семьи – это увлекательное и значимое занятие, позволяющее углубить понимание личных судеб в свете исторических событий. В последние годы доступность онлайн-ресурсов значительно упростила этот процесс, особенно в отношении периода Великой Отечественной войны. Она является одним из наиболее изученных периодов, благодаря множеству созданных в то время документов и баз данных, которые теперь доступны в интернете. Однако, поиск информации о более ранних событиях часто требует обращения в архивы и может быть затруднен из-за отсутствия полных данных. Выбор периода Великой Отечественной войны для реконструкции истории семьи объясняется несколькими причинами. Во-первых, сохранилось огромное количество документов, многие из которых были оцифрованы и выложены в сеть. Во-вторых, участие миллионов людей в войне облегчает поиск информации о предках через различные онлайн-базы данных. В-третьих, изучение этого периода позволяет лучше понять роль семьи в контексте значимых исторических событий. Для реконструкции истории семьи можно использовать различные онлайн-ресурсы, которые можно условно разделить на общие базы данных и архивы. Общие базы данных: ОБД "Мемориал" [1], Память народа [2], Подвиг народа [3]. Архивы и специализированные ресурсы: Центральный архив Министерства обороны Российской Федерации (ЦАМО РФ) [4], Национальный архив Республики Беларусь [5], Региональные архивы Беларуси [6]. В рамках данной статьи была проведена реконструкция истории семьи, включающая поиск и анализ информации о предках, живших во время Великой Отечественной войны. Этот опыт демонстрирует, как онлайн-ресурсы и архивы могут помочь в восстановлении семейной истории. Иван Степанович Лебедь, прадед моего отца по материнской линии, родился в 1924 г. в деревне Быстрица, Копыльского района Минской области (БССР) [1,2]. По национальности он был белорусом [4]. С 01.10.1943 по 01.08.1944 Иван Степанович служил в партизанских формированиях [4,5]. Он входил в состав 32-й партизанской бригады имени М. И. Калинина, а непосредственно его отряд носил имя А. Я. Пархоменко [4,6]. В рядах партизан он занимал должность рядового бойца [4]. За свои заслуги Иван Степанович был награжден медалью «Партизану Отечественной войны II степени», Орденом Красной Звезды и Орденом Партии Ленина-Сталина [2-3]. Среди его подвигов: уничтожение 7 километров линии связи, подрыв 6 мостов, 4 автомобилей противника (убито 3 немца), 2 повозок с немецкими солдатами и 12 рельсов железнодорожного полотна. Благодаря его действиям было ранено около 50 врагов [4,5]. Иван Сергеевич Пря-

моносов, дед моей матери по материнской линии, родился 9 июля 1918 г. Его родиной является Курганская область: село Сосново Ольховского района или село Сосновское Шадринского района (сведения разнятся) [1,2]. 14 сентября 1938 г. он был призван на службу Свердловским городским военным комиссариатом (ГВК) [1,2]. За годы службы И. С. Прямоносов дослужился до звания майора, а также занимал должности младшего лейтенанта и лейтенанта [2,4]. Он служил в различных воинских частях, включая 790-й истребительный авиационный полк и 26-й отдельный авиационный полк гражданского воздушного флота [2,4]. Его боевые заслуги отмечены множеством наград: медалью «За боевые заслуги», медалью «За победу над Германией в Великой Отечественной войне 1941–1945 гг.», Орденом Красного Знамени, Орденом Красной Звезды, медалью «За взятие Кёнигсберга», а также Орденом Отечественной войны I и II степени [2,3]. Борис Алексеевич Полин, брат матери моего деда по материнской линии, родился в Москве в 1924 году [1,2]. На военную службу он был призван 10 августа 1942 г. Таганским районным военным комиссариатом Москвы [1,2]. Сержант Б. А. Полин проходил службу в различных подразделениях: пересыльных пунктах Московского городского военного комиссариата и Владимирского военного пехотного училища, а также в составе 44-й гвардейской стрелковой дивизии. Его воинское звание было сержант и призывник [2,4]. Петр Поликарпович Кузнецов, прадед по материнской линии (отец моего деда), родился 25 июня 1922 г. в Белорусской ССР, Витебской области. Его родным местом называют Сенненский район — деревни Гулино, Пустынки или Теолиново (источники указывают разные данные) [1,2]. На службу он был призван Сенненским районным военным комиссариатом дважды — сначала 9 сентября 1940 г., затем повторно 27 сентября того же года [1-2]. За годы службы П. П. Кузнецов дослужился до звания подполковника; также занимал звания капитана и лейтенанта [2,4]. Он служил в 5-й тыловой бригаде РГК, 21-й гвардейской стрелковой дивизии, Прибалтийском военном округе, Резерве Главного командования, 46-м запасном стрелковом полку и 288-й стрелковой дивизии [2,4]. Боевые заслуги Петра Поликарповича Кузнецова отмечены Орденом Красного Знамени, медалью «За боевые заслуги», медалью «За победу над Германией в Великой Отечественной войне 1941–1945 гг.» и Орденом Красной Звезды [2,3]. Реконструкция истории семьи позволяет восстановить утраченные детали биографий предков и оценить их вклад в исторические события. Эта информация была получена из различных ресурсов, включая ОБД "Мемориал" "Память народа" "Подвиг народа" ЦАМО РФ, региональные архи-

вы и Национальный архив Республики Беларусь, исторические справочники и музеи, такие как Музей Победы. При поиске информации о предках часто возникают следующие сложности: ограниченность данных в некоторых архивных системах может препятствовать полной реконструкции истории семьи. Некоторые документы могут быть утеряны или не оцифрованы, что требует обращения в архивы напрямую. Необходимость физических поездок в архивы для получения полной информации о предках до периода Второй мировой войны также является распространенной проблемой. Многие исторические документы до сих пор хранятся в бумажном виде и доступны только в архивах. Кроме того, проблемы с доступностью архивов и их состоянием также могут усложнить процесс поиска. Некоторые архивы могут иметь ограниченные часы работы или требовать специальных разрешений для доступа к документам. Реконструкция истории семьи через онлайн-системы позволяет глубже понять личные судьбы в контексте исторических событий. Использование специализированных ресурсов и архивов может существенно облегчить этот процесс. Однако для более полной картины часто требуется

обращение к архивам и прямое общение с родственниками. Важно сохранять память о людях, участвовавших в Великой Отечественной войне, и их вклад в победу. Реконструкция семейной истории помогает не только понять прошлое, но и оценить роль семьи в контексте крупных исторических событий.

Список литературы

1. Суворов А. М. Брестская крепость. Война и мир. / А. М. Суворов. – Брест : Полиграфика, 2010. – 255 с.
2. Национальный мемориальный комплекс «Брестская крепость-герой» : [Электронный ресурс]. – Режим доступа: <https://brest-fortress.by/novosti/614-50-let-postupamyati-1>. – Дата доступа: 05.04.2025.
3. Лесин А. А. Музеефикация фортов Брестской крепости: достижения в коммеморации отечественной военной истории // Культура. Наука. Творчество = Культура. Наука. Творчасць = Culture. Science. Arts. – 2022. – С. 163-167.
4. Наш город. Брест в деталях: Пост Памяти Брестской крепости (авт. и вед. И. Альшова; гость И. Чикалов): [Электронный ресурс]. – Режим доступа: <https://www.youtube.com/watch?v=jIYTLm03LgE>. – Дата доступа: 06.04.2025.
5. Попов А. Н. Мемориальная культура и историческая память: традиции и трансформации. – М.: Изд-во РГГУ, 2018. – С. 145.

Хаджинова К.А., студентка 1 курса ФИТУ БГУИР.

Научный руководитель: Сугако Наталья Анатольевна, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

ВИЦЕ-АДМИРАЛ ИЗ БУДА-КОШЕЛЁВО

Валентин Петрович Дрозд родился в сентябре 1906 г. на станции Буда-Кошелёвская Гомельской области. Он с детства мечтал стать моряком, но по совету отца идёт работать подручным слесаря в железнодорожные мастерские. В 1924 г. он приезжает в Ленинград, устраивается на Кировский завод, а через год по комсомольской путёвке поступает в Высшее военно-морское училище имени М.В. Фрунзе. После окончания училища в 1928 г. В.П. Дрозд служил на кораблях Краснознамённого Балтийского флота. В 1936-1940 гг. он сражался на стороне республиканской Испании, за что был награждён орденами Ленина и Красного Знамени. [1]. С мая 1938 г. по июль 1940 г. командовал Северным флотом. В сентябре 1940 г. он был назначен на должность начальника Черноморского высшего военно-морского училища, а в феврале 1941 г. – командиром отряда лёгких сил [2]. С первых дней Великой Отечественной войны В.П. Дрозд находился на боевых кораблях. За умелое проведение боевых операций 16 сентября 1941 г. ему было присвоено воинское звание «вице-адмирал» [3]. В этом же месяце он был назначен на должность командующего эскадрой кораблей Краснознамённого Балтийского флота. Под его командованием флот с минимальными потерями перешёл из Таллина в Кронштадт для обороны города на Неве. Дальнобойная артиллерия кораблей Балтийского флота помогла Красной Армии сдержать наступление немецко-фашистских войск на Ленинград. Особенно отличился наш земляк в переброске героических защитников Ханко на Ленинградский фронт. Вице-адмирал В.П. Дрозд разработал план, который лично претворил в жизнь: в первую ночь идти до острова Гогланд, во вторую – до полуострова Ханко. Вот как описывал картину похода участник тех событий В. Галичников: «...Ветер слепит глаза, ледяные брызги летят на палубу. Стынут ноги, покрываются инеем ресницы, немеют руки, но люди не уходят с постов. В. П.

Дрозд вместе со всеми делит трудности тяжелейшего похода... Ханковские походы вице-адмирала Дрозда вошли в историю Краснознамённого Балтийского флота как её героические страницы. Он был отважным человеком ... Моряки безбоязненно шли за ним в самые рискованные операции» [4]. Указом Президиума Верховного Совета СССР от 17 января 1942 г. В.П. Дрозд был награждён орденом Красного Знамени. 29 января 1943 г. он трагически погиб при исполнении боевого задания [5]. Адмиралу было 37 лет. Почти два года адмиральской службы он провёл в сражениях, руководя сложными боевыми операциями. Указом Президиума Верховного Совета СССР от 13 февраля 1943 г. гвардейский эскадренный миноносец «Стойкий» Краснознамённого Балтийского флота был переименован в гвардейский эскадренный миноносец «Вице-адмирал Дрозд». Почему именно этот миноносец? Дело в том, что все боевые операции, за которые «Стойкий» удостоился гордого звания «гвардейский», проходили под командованием В.П. Дрозда. В экспозиции Белорусского государственного музея истории Великой Отечественной войны представлены: фотопортрет вице-адмирала В. П. Дрозда, его фуражка, благодарственное письмо и блокнот, подаренный ему экипажем эсминца «Стойкий». В городе Буда-Кошелёво Гомельской области одна из улиц носит имя вице-адмирала Валентина Петровича Дрозда.

Список литературы

1. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. НД 10900.
2. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. НД 10900.
3. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. НД 10900 1.
4. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. НД 9198 5 5.
5. Белорусский государственный музей истории Великой Отечественной войны. – Фонды. НД 10900 1.

Цуца З. Р., студентка 1 курса ФИТУ БГУИР

Маслихин М. В., студент 1 курса ФИТУ БГУИР

Научный руководитель: Сугако Наталья Анатольевна, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

Подсекция «Политология»

Председатель:	канд. ист. наук, доцент Николаева Л. В.
Члены жюри:	канд. ист. наук, доцент Вашкевич И. В. преподаватель Янковский Ю. Ю.
Секретарь	канд. ист. наук, доцент Сугако Н. А. преподаватель Галицкая Е. М.

ВЛИЯНИЕ ЭМИГРАЦИИ НА ФОРМИРОВАНИЕ ПОЛИТИЧЕСКИХ ВЗГЛЯДОВ В.И. ЛЕНИНА

Владимир Ильич Ленин (1870-1924) – выдающийся российский революционер, политик и мыслитель, автор концепции диктатуры пролетариата. Он возглавлял большевистскую партию и сыграл решающую роль в свершении Октябрьской революции 1917 г., адаптировав марксистские идеи к российским условиям. Свою революционную деятельность он начал, вступив в 1888 г. в марксистский кружок в Казани. Здесь на молодого революционера оказали большое влияние работы Г.В. Плеханова: после их изучения взгляды В. Ульянова стали смещаться от народнических к социал-демократическим. В.И. Ленин активно пропагандировал марксистские идеи через свои статьи и публичные выступления. Он видел в марксизме основу для революционной борьбы и социальных преобразований в России. В 1893 г. В. Ульянов переехал в Санкт-Петербург, где трудился на юридическом поприще. В 1895 г. во время заграничной поездки он лично познакомился с отцами-основателями европейского социал-демократического движения: В. Либкнехтом, П. Лафаргом и Г. Плехановым. По возвращении на родину В. Ульянов приложил массу усилий к тому, чтобы объединить разрозненные петербургские марксистские кружки. Результатом проделанной работы стал «Союз борьбы за освобождение рабочего класса», который он возглавил. Целью провозглашалось свержение самодержавия. В декабре 1895 г. участники союза были арестованы за пропаганду революционных идей. Более года он провёл в заключении. В 1897 г. был сослан на три года в село Шушенское Минусинского округа Енисейской губернии. Владимир Ильич написал свыше тридцати произведений. В них он наметил задачи русских социал-демократов, обосновал необходимость создания пролетарской партии, разработал ее программу и тактику. Наиболее значительным произведением В.И. Ленина в этот период была книга «Развитие капитализма в России». В этой книге он доказал, что капитализм укреплялся как в промышленности, так и в сельском хозяйстве, вскрыл глубочайшие противоречия капиталистического общества, обосновал его руководящую роль в революционном движении, подчеркнул необходимость союза пролетариата с крестьянством. Из Сибири ему удалось наладить связь с марксистами Санкт-Петербурга, Москвы и других городов, среди которых он приобрёл известность под псевдонимом К. Тулин. В июле 1900 г. В.И. Ленин выехал из России в Германию, чтобы там организовать издание общерусской марксистской газеты. Его путь лежал через Швейцарию. С организации общерусской марксистской газеты началось осуществление ленинского плана построения самостоятельной партии пролетариата. Дело создания «Искры» стави-

лось с учетом опыта всей демократической и социалистической печати прошлого. В новых исторических условиях, спустя несколько десятилетий после «Полярной Звезды» и «Колокола», со страниц «Искры» зазвучало могучее слово российской революционной прессы. За полтора года вышел 21 номер «Искры», налажен выпуск журнала «Заря» и многих других партийных изданий. В декабре 1901 г. В журнале «Заря» В.И. Ульянов впервые выступил в печати под псевдонимом Ленин. Сплотив вокруг «Искры» партийные организации России, В.И. Ленин и его соратники развернули подготовку ко II съезду партии. В этот же период впервые был употреблен термин "ленинизм придуманный одним из основоположников меньшевизма Юлием Осиповичем Мартовым. В середине декабря 1905 г., в финском городе Тампере состоялась первая партийная конференция большевиков. В. И. Ленин был избран председателем конференции и выступил перед ее участниками с докладами и по текущему моменту. Большую часть 1908 г. он проводит в Женеве. Активно переписывается с единомышленниками, работает над собранием сочинений, докладами и статьями для «Пролетария», позже выступает с рефератами в Париже и Лозанне. Из Парижа уезжает в Лондон для работы над книгой «Материализм и эмпириокритицизм». В течение лета проводит несколько дней в горах в местечке Вер Леглиз (Швейцария), а в конце года перебирается из Женевы в Париж. В октябре 1908 г. В.И. Ленин посещает Льеж, где читает рефераты. Затем выезжает из Парижа в Брюссель на одиннадцатую сессию Международного социалистического бюро (МСБ). Часто работает в парижской Национальной библиотеке. В 1909 г. Ленин опубликовал свой главный философский труд «Материализм и эмпириокритицизм», в котором он критиковал эмпириокритицизм и махизм, которые получили популярность среди социал-демократов. Этот труд стал важным шагом в формировании марксистской философии и укреплении позиций большевиков в теоретической борьбе с другими фракциями. Помимо философских работ, В.И. Ленин продолжал писать статьи и брошюры, развивая свои теоретические взгляды на социализм и революцию. Он критиковал различные течения внутри РСДРП, такие как ликвидаторство и отзовизм, стремясь сохранить революционную направленность партии. В.И. Ленин активно занимался редакторской работой, публикуя различные издания, включая газеты и журналы, которые распространялись среди русской эмиграции и в России. На совещании расширенной редакции газеты «Пролетарий» в июне 1909 г. произошло размежевание большевиков с отзовистами, ультиматистами и махистами, что стало важным шагом в консолидации

большевистской фракции. В.И. Ленин участвовал в международных социалистических конгрессах и собраниях, где отстаивал свои идеи и влиял на развитие социалистического движения в Европе. Как представитель РСДРП во II Интернационале, Ленин стремился укрепить позиции большевиков на международной арене и сформировать единую линию в отношении войны и революции. В 1911 г. В.И. Ленин основал партийную школу в парижском пригороде Лонжюмо. Она стала важным центром для обучения и идеологической подготовки будущих лидеров большевистской партии. В.И. Ленин активно критиковал других лидеров социал-демократии, таких как Л. Троцкий и меньшевики, стремясь укрепить позиции большевиков. Он боролся с отзовистами и ультиматистами — радикальными большевиками, выступавшими против участия в работе Государственной Думы. Эта борьба помогла Ленину сохранить контроль над большевистской фракцией и обеспечить ее единство перед лицом внутренних и внешних вызовов. С началом Первой мировой войны В.И. Ленин переехал из Парижа в Цюрих, где продолжал свою политическую и теоретическую работу. В Швейцарии он продолжал развивать свои идеи о войне и революции, что стало основой для его знаменитой работы «Империализм, как высшая стадия капитализма». Этот переезд позволил Ленину сохранить связь с европейским социалистическим движением и подготовить почву для его возвращения в Россию в 1917 г. В Цюрихе, за-

ручившись поддержкой Ф. Платтена, В.И. Ленин смог получить вид на жительство. Он продолжал посещать все собрания интернационалистской организации «Эйнтрахт», был на партийных собраниях местных социалистов, поскольку являлся членом Социал-демократической партии Швейцарии. В.И. Ленин, находясь в подполье, посылает свое последнее письмо из Финляндии членам ЦК большевиков с указанием на необходимость немедленного захвата власти. В.И. Ленин попал в Петроград в октябре 1917 г. после подготовки его возвращения из Финляндии. Он заранее отправил просьбу о подготовке удостоверения на имя Константина Петровича Иванова, чтобы получить доверие и поддержку со стороны местных Советов, включая Выборгский Совет солдатских депутатов. 16 октября 1917 г. в помещении Лесовско-Удельнинской районной думы на Выборгской стороне (Петроград) происходило расширенное заседание Центрального Комитета (ЦК) большевистской партии. Это заседание было критически важно для организации и подготовки вооруженного восстания, которое должно было привести к захвату власти. Таким образом, эмиграция оказала значительное влияние на формирование политических взглядов Владимира Ильича Ленина, способствуя его теоретическому развитию, укреплению связей с международным социалистическим движением, а в дальнейшем провозглашении власти социализма в Российской Империи.

Земляник Р.В., студентка 1 курса ФИТУ БГУИР.

Научный руководитель: Сугако Наталья Анатольевна, доцент кафедры гуманитарных дисциплин, кандидат исторических наук, доцент

Содержание

<i>Секция «Автоматизированные системы обработки информации»</i>	4
А. А. Агель, М. В. Пашковец, Н. В. Хаджинова СРАВНИТЕЛЬНЫЙ АНАЛИЗ ФРЕЙМВОРКОВ ДЛЯ РАЗРАБОТКИ MAS	5
П. А. Арсенович, Р. В. Земляник АВТОМАТИЗИРОВАННАЯ СИСТЕМА ГЕНЕРАЦИИ ВОПРОСОВ	6
С. А. Беспалов ПРИНЯТИЕ РЕШЕНИЙ И ТЕОРИЯ ИГР. ОСНОВНЫЕ ПОНЯТИЯ, ПРОБЛЕМЫ РЕАЛИЗАЦИИ	7
В. В. Василевский Р. Д. Рында АЛГОРИТМ ИЗМЕНЕНИЯ ЭМОЦИОНАЛЬНОЙ ОКРАСКИ АУДИО-ФАЙЛА	8
А. К. Грабовский АВТОМАТИЧЕСКОЕ РАСПОЗНАВАНИЕ АВТОМОБИЛЬНЫХ НОМЕРОВ	9
М. А. Иосько ОПТИМИЗАЦИЯ РАБОТЫ КОПИРОВАЛЬНОГО ЦЕНТРА	10
В. А. Кадлубай ЭТИЧЕСКИЕ АСПЕКТЫ ПРИНЯТИЯ РЕШЕНИЙ В АВТОНОМНЫХ MAS	11
V. A. Kadlubai DATA CLEANING: CHALLENGES AND MODERN APPROACHES	12
Д. В. Кононцева, А. А. Мельник ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ В ЗАКАЗНОЙ РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	13
Д. В. Кононцева, А. А. Мельник ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ПРОДУКТОВОЙ РАЗРАБОТКЕ	14
Д. В. Коршикова, Е. В. Купчина АЛГОРИТМ МАСШТАБИРОВАНИЯ ГРАФИЧЕСКИХ ЭЛЕМЕНТОВ: ПРИНЦИПЫ И ПРИМЕРЫ	15
Е. В. Купчина, Д. В. Коршикова АЛГОРИТМ ВОССТАНОВЛЕНИЯ ИЗОБРАЖЕНИЙ: ПРИНЦИПЫ И ПРИМЕРЫ	17
А. И. Михнюк, П. С. Савчиц ДИНАМИЧЕСКОЕ РАЗРЕШЕНИЕ КОНФЛИКТОВ В ГЕТЕРОГЕННЫХ MAS НА ОСНОВЕ ТЕОРИИ ИГР С НЕПОЛНОЙ ИНФОРМАЦИЕЙ	19
А. О. Прохор, В. И. Ярмолик СИСТЕМЫ УПРАВЛЕНИЯ РЕСУРСОЕМКИМИ КОНТЕЙНЕРИЗИРОВАННЫМИ ПРИЛОЖЕНИЯМИ	20
М. О. Савоневская, В. В. Столярова, Л. А. Черникова СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ СЛОВ В ВЕКТОРНЫХ БАЗАХ ДАННЫХ	22

М. О. Савоневская, В. В. Столярова, Л. А. Черникова ИСПОЛЬЗОВАНИЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ (RL) ДЛЯ ПРЕДОТВРАЩЕНИЯ ТУПИКОВЫХ СИТУАЦИЙ В СИСТЕМАХ С ОБЩИМИ РЕСУРСАМИ	23
К. Д. Сморщок, Д. Д. Хмыз СИСТЕМА АВТОМАТИЧЕСКОГО РАСПОЗНАВАНИЯ ДОРОЖНЫХ ЗНАКОВ	24
В. Ю. Служалик ПОСТРОЕНИЕ ДЕРЕВА КРАТЧАЙШИХ РАССТОЯНИЙ МИНИ- МАЛЬНОЙ СТОИМОСТИ	25
А. А. Столбанов АЛГОРИТМ РАСЧЕТА ДОХОДНОСТИ ВАЛЮТНОГО ПОРТФЕЛЯ	26
А. А. Струкова АЛГОРИТМ ВЗАИМОДЕЙСТВИЯ КОМПОНЕНТОВ НА СТОРОНЕ КЛИЕНТА	28
К. А. Хаджинова ИСПОЛЬЗОВАНИЕ BASH СКРИПТОВ ДЛЯ РАБОТЫ С ОПЕРАЦИ- ОННОЙ СИСТЕМОЙ WINDOWS	29
М. А. Хващевский, Е. В. Протченко ИСПОЛЬЗОВАНИЕ МЕТОДОВ ЭКСПЕРТНЫХ ОЦЕНОК ПРИ ПО- СТУПЛЕНИИ В ВУЗ	30
<i>Секция «Интеллектуальные информационные технологии»</i>	33
Али Марван ГИБРИДНАЯ ОПТИМИЗАЦИЯ КАРПУЛИНГОВЫХ СИСТЕМ ДЛЯ УСТОЙЧИВОЙ ГОРОДСКОЙ МОБИЛЬНОСТИ	34
Али Марван СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПЛАТФОРМЫ OSTIS И ПОПУЛЯР- НЫХ ФРЕЙМВОРКОВ ДЛЯ РАЗРАБОТКИ ИИ-АГЕНТОВ	35
Т. М. Робилко МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ЭЛЕКТРОКАРДИО- ГРАММ	37
Г. А. Славинский ПРИМЕНЕНИЕ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ВОС- СТАНОВЛЕНИЯ ЗАШУМЛЕННОЙ ИНФОРМАЦИИ	39
Н. А. Шерстобитов, А. В. Зинькович МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО ПОИСКА, АДАПТИВНОЙ СОР- ТИРОВКИ И РЕКОМЕНДАЦИЙ НА МУЗЫКАЛЬНЫХ СТРИМИНГОВЫХ СЕРВИ- САХ	44
<i>Секция «Вычислительные методы и программирование»</i>	47
А. В. Акименко ВИЗУАЛИЗАЦИЯ ТРЁХМЕРНЫХ СЦЕН МЕТОДОМ ТРАССИРОВКИ ПУТИ С ИСПОЛЬЗОВАНИЕМ ГРАФИЧЕСКОГО API VULKAN	48
Б. К. Адамович СРАВНЕНИЕ ГРАФИЧЕСКОГО РЕНДЕРИНГА НА CPU, CUDA И RT- ЯДРАХ	50

Т. А. Авдеева МАТЕМАТИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ПРОГРЕССИИ ПЕРСОНА- ЖА	52
Я. С. Баранова, Д. А. Рутковская ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ФИЛЬТРОВ НА ДАЛЬТОНИЗМ	54
А. А. Бондарук Д. С. Харланчук СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ ГЛОБАЛЬНОГО ОСВЕ- ЩЕНИЯ (PATH TRACING, PHOTON MAPPING, V-RAY)	56
В. Д. Бубен ЭНТРОПИЯ КАК МЕТРИКА ОЦЕНКИ ПОЛИГОНАЛЬНЫХ СЕТОК	57
К. Г. Бузенкова, Я. Г. Кривцов ВЛИЯНИЕ ПАРАМЕТРОВ СИСТЕМ ЧАСТИЦ НА ВРЕМЯ РЕНДЕ- РИНГА В 3DS MAX	58
А. К. Бурая СЕЗОННОСТЬ В ИГРОВОЙ ИНДУСТРИИ СНГ: АНАЛИЗ ПРОДАЖ И ПОТРЕБИТЕЛЬСКОГО СПРОСА	59
М. В. Голенчук МАТЕМАТИЧЕСКИЕ ПРИНЦИПЫ И АЛГОРИТМЫ МОРФИНГА В 3DS MAX	60
А. А. Голодко Я. Р. Сахацкая ПРИМЕНЕНИЕ МАШИННОГО ОБУЧЕНИЯ ДЛЯ АВТОМАТИЗА- ЦИИ СОЗДАНИЯ ТРЕХМЕРНЫХ АНИМАЦИЙ ИЗУЧЕНИЕ ВОЗ- МОЖНОСТЕЙ НЕЙРОННЫХ СЕТЕЙ ДЛЯ ГЕНЕРАЦИИ АНИМА- ЦИЙ ИЛИ СОЗДАНИЯ РЕАЛИСТИЧНЫХ ДВИЖЕНИЙ	61
А. В. Галушко АЛГОРИТМ ГЕНЕРАЦИИ ЛАНДШАФТА UNITY	62
Т. С. Киселёва, О. А. Шеремет МЕТОДЫ РЕТОПОЛОГИИ ПОЛИГОНАЛЬНОЙ СЕТКИ	64
А. В. Козлов СОЗДАНИЕ АНИМАЦИИ ЛИЦА	65
А. В. Козлов, А. С. Макаревич ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ В КОМПЬЮТЕРНЫХ ИГРАХ	67
Корытко З. С., Кушель К. О. ИСПОЛЬЗОВАНИЕ ДИАГРАММ ВОРОНОГО И ТРИАНГУЛЯЦИИ ДЕЛОНЕ ДЛЯ СОЗДАНИЯ ТРЁХМЕРНЫХ ИГРОВЫХ ЛАНДШАФ- ТОВ	69
К. О. Кушель КОМЬЮНИТИ-МЕНЕДЖМЕНТ И ЕГО РОЛЬ В МАРКЕТИНГОВОЙ СТРАТЕГИИ ИГРОВОГО ПРОДУКТА	71
А. О. Линкевич РАЗРАБОТКА ЧАТ-БОТА ДЛЯ АВТОМАТИЗИРОВАННОГО ПОЛУ- ЧЕНИЯ РАСПИСАНИЯ	73
И. А. Пименов FRAMEWORK RPGBUILDER	74
И. А. Пименов, А. Д. Шкадинский ВЛИЯНИЕ VR И AR НА ИНДУСТРИЮ ИГР	76

Г. Д. Радьков, А. С. Беликов ФИЗИЧЕСКИ КОРРЕКТНОЕ ОСВЕЩЕНИЕ В 3D-ГРАФИКЕ: МАТЕМАТИЧЕСКИЕ МОДЕЛИ И АЛГОРИТМЫ	78
Д. Д. Рязанцев, Н. Д. Рязанцев ОСОБЕННОСТИ РАЗРАБОТКИ АЛГОРИТМА А* В ИГРОВOM ПРИЛОЖЕНИИ	80
Н. Д. Рязанцев, Д. Д. Рязанцев МЕТОДЫ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ДАННЫХ ДЛЯ ОПТИМИЗАЦИИ ОБУЧЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ	82
Н. А. Шерстобитов, А. В. Зинькович МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО ПОИСКА, АДАПТИВНОЙ СОРТИРОВКИ И РЕКОМЕНДАЦИЙ НА МУЗЫКАЛЬНЫХ СТРИМИНГОВЫХ СЕРВИСАХ	84
В. О. Царюк, А. Л. Макаревич ВЛИЯНИЕ КОЛИЧЕСТВА ПОЛИГОНОВ НА КАЧЕСТВО РЕНДЕРА В BLENDER CYCLES И ARNOLD ДЛЯ 3DS MAX	86
<i>Секция «Электрические цепи»</i>	87
Я. В. Кипцевич СИСТЕМА АКТИВНОЙ НАВИГАЦИИ МОБИЛЬНОГО РОБОТА	88
Я. В. Кипцевич МОДУЛЬ КОММУНИКАЦИОННОЙ ПЛАТФОРМЫ ДЛЯ БЕСПРОВОДНОГО ИНФОРМАЦИОННОГО ВЗАИМОДЕЙСТВИЯ С БОРТОВЫМИ СИСТЕМАМИ МОБИЛЬНЫХ МАШИН	90
Е. А. Ламака, З. И. Лешик ПОЛНОМОСТОВОЙ DC-DC ПРЕОБРАЗОВАТЕЛЬ С ФАЗОВЫМ СДВИГОМ	92
П. О. Смирнов, А. А. Сохин МЕТОД ЭФФЕКТИВНОГО ЗАРЯДА НИКЕЛЬ-МЕТАЛЛГИДРИДНЫХ АККУМУЛЯТОРОВ	94
П. М. Ясинский, И. В. Потёмин ПРОЕКТИРОВАНИЕ РАЗДЕЛИТЕЛЬНЫХ ФИЛЬТРОВ ДЛЯ МНОГОПОЛОСНОЙ СИСТЕМЫ ВОСПРОИЗВЕДЕНИЯ	96
<i>Секция «Гуманитарные дисциплины»</i>	97
И. В. Акуленко ИСТОРИЯ РАЗВИТИЯ БЕЛОРУССКОГО ТЕЛЕРАДИОВЕЩАНИЯ	98
И. А. Большов ИСТОРИЯ ТРАНСПЛАНТАЦИИ НА БЕЛАРУСИ	99
В. В. Василевский В. А. Микша ОТНОШЕНИЕ ЦЕРКВИ, ГОСУДАРСТВА И ОБЩЕСТВА К ВНЕБРАЧНЫМ СВЯЗЯМ В ВЕЛИКОМ КНЯЖЕСТВЕ ЛИТОВСКОМ	100
Е. Д. Волощик НАСЛЕДИЕ РОДА ПУСЛОВСКИХ КАК ОТРАЖЕНИЕ КУЛЬТУРНЫХ И ИСТОРИЧЕСКИХ ПРОЦЕССОВ XIX ВЕКА	101

А. С. Гаркуша К. В. Данілава ГІСТАРЫЧНАЕ МІНУЛАЕ Ў РАМАНАХ УЛАДЗІМІРА КАРАТКЕВІ- ЧА	102
А. А. Забаровский ЖИЗНЕННЫЙ ПУТЬ РУДЕНКОВА НИКОЛАЯ ФЁДОРОВИЧА И РУ- ДЕНКОВОЙ КЛАВДИИ ФЕДОТОВНЫ	103
Е. Д. Зубович РАСКОПКИ НА Р. МЕНКЕ: НОВЫЕ ОТКРЫТИЯ	104
Д. А. Новиков ВИЛЕНСКО-РАДОМСКАЯ УНИЯ 1401 Г.: ИТОГИ И ПОСЛЕДСТВИЯ	105
Ю. М. Павлюченко РАЗВИТИЕ ЖЕЛЕЗНОЙ ДОРОГИ РЕСПУБЛИКИ БЕЛАРУСЬ	106
Т. Л. Федорин ЖИЗНЕННЫЙ ПУТЬ И ВОСПОМИНАНИЯ О ВЕТЕРАНЕ ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ ПИРОГОВСКОМ Е. В.	109
А. А. Яцков СОЗДАНИЕ ЗАПАДНОЙ ОБЛАСТИ – ЗАПАДНОЙ КОММУНЫ КАК ПРЕДШЕСТВЕННИЦЫ ССРБ	111
П. А. Арсенович ПОСТ №1 У ВЕЧНОГО ОГНЯ БРЕСТСКОЙ КРЕПОСТИ-ГЕРОЙ	113
А. Д. Вильтовская МОЙ ПРАДЕД – ГЕРОЙ!	115
Е. В. Дмитриевич НИКОЛАЙ ЕГОРОВИЧ БОРАНЕНКОВ: ВОЕННЫЙ ПУТЬ ПИСАТЕ- ЛЯ	116
Д. А. Двораковский А. Д. Харкевич БОЕВОЙ ПУТЬ МАРАТА ИВАНОВИЧА КАЗЕЯ В ПЕРИОД ВЕЛИ- КОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ	117
К. П. Ефимчик ТРОСТЕНЕЦ – СИМВОЛ ТРАГЕДИИ В МОЕЙ СЕМЬЕ	118
Д. С. Ивашевич ИСПОЛЬЗОВАНИЕ ТРУДА УЗНИКОВ В КОНЦЛАГЕРЯХ ГЕРМА- НИИ	119
К. П. Концевич ФЕНОМЕН СТАТИСТИКИ НЕМЕЦКИХ И СОВЕТСКИХ ЛЕТЧИКОВ- АСОВ ПЕРИОДА ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ	121
К. А. Пейганович СУДЬБА МОЕГО ПРАДЕДА ВО ВРЕМЯ ВЕЛИКОЙ ОТЕЧЕСТВЕН- НОЙ ВОЙНЫ	123
А. Д. Перко ОСВОБОЖДЕНИЕ НОВОГРУДКА В ИЮЛЕ 1944 Г. ПО РАССКАЗАМ ОЧЕВИДЦЕВ	124
Я. И. Пинчук ЛЕТОПИСЬ ВОЙНЫ В МОЕЙ СЕМЬЕ	125

К. А. Попов ГЕНОЦИД НА ТЕРРИТОРИИ ЛЮБАНСКОГО РАЙОНА В ГОДЫ ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ	127
В. Ю. Пракаповіч П. М. Ясінскі ГЕНАЦЫД БЕЛАРУСКАГА НАРОДА (ДОКШЫЦКІ РАЁН)	129
К. А. Хаджинова РЕКОНСТРУКЦИЯ ИСТОРИИ СЕМЬИ ЧЕРЕЗ ПОИСКОВЫЕ ОНЛАЙН СИСТЕМЫ	130
З. Р. Цупа М. В. Маслихин ВИЦЕ-АДМИРАЛ ИЗ БУДА-КОШЕЛЁВО	132
Р. В. Земляник ВЛИЯНИЕ ЭМИГРАЦИИ НА ФОРМИРОВАНИЕ ПОЛИТИЧЕСКИХ ВЗГЛЯДОВ В.И. ЛЕНИНА	134

Научное издание

**61-Я НАУЧНАЯ КОНФЕРЕНЦИЯ АСПИРАНТОВ, МАГИСТРАНТОВ
И СТУДЕНТОВ УЧРЕЖДЕНИЯ ОБРАЗОВАНИЯ
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»**

(МИНСК, 21–25 АПРЕЛЯ 2025 ГОДА)

МАТЕРИАЛЫ КОНФЕРЕНЦИИ
по направлению 2:

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И УПРАВЛЕНИЕ

авторской редакции

Ответственный за выпуск *М. В. Ковалёв*

Компьютерная верстка *М. В. Ковалёв*

Дизайн обложки *А. А. Навроцкий*

Подписано в печать 16.05.2025. Формат 60×84 ¹/₈. Бумага офсетная.
Гарнитура «Computer Modern». Отпечатано на ризографе. Усл. печ. л.
10,70 Уч.-изд. л. 25,1. Тираж экз. Заказ 76.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники».

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий №1/238 от 24.03.2014,
№2/113 от 07.04.2014, №3/615 от 07.04.2014.

Ул. П. Бровки, 6, 220013, г. Минск