

www.aurora-group.eu
www.nbpublish.com

ПРОГРАММНЫЕ СИСТЕМЫ

И

ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ

научный журнал



Выходные данные

Номер подписан в печать: 03-04-2025

Учредитель: Даниленко Василий Иванович, w.danilenko@nbpublish.com

Издатель: ООО <НБ-Медиа>

Главный редактор: Морозов Михаил Николаевич, кандидат технических наук,
mikhail.n.morozov@gmail.com

ISSN: 2454-0714

Контактная информация:

Выпускающий редактор - Зубкова Светлана Вадимовна

E-mail: info@nbpublish.com

тел.+7 (966) 020-34-36

Почтовый адрес редакции: 115114, г. Москва, Павелецкая набережная, дом 6А, офис 211.

Библиотека журнала по адресу: http://www.nbpublish.com/library_tariffs.php

Publisher's imprint

Number of signed prints: 03-04-2025

Founder: Danilenko Vasiliy Ivanovich, w.danilenko@nbpublish.com

Publisher: NB-Media ltd

Main editor: Morozov Mikhail Nikolaevich, kandidat tekhnicheskikh nauk,
mikhail.n.morozov@gmail.com

ISSN: 2454-0714

Contact:

Managing Editor - Zubkova Svetlana Vadimovna

E-mail: info@nbpublish.com

тел.+7 (966) 020-34-36

Address of the editorial board : 115114, Moscow, Paveletskaya nab., 6A, office 211 .

Library Journal at : http://en.nbpublish.com/library_tariffs.php

Редакционный совет

Гельман Виктор Яковлевич – доктор технических наук, профессор, профессор кафедры медицинской информатики и физики ФГБОУ ВО «Северо-Западный государственный медицинский университет им. И.И.Мечникова», 191015, Россия, г. Санкт-Петербург, ул. Кирочная, д.41, gelm@sg2104.spb.edu

Поляков Виктор Павлович – доктор педагогических наук, профессор, Главный научный сотрудник лаборатории психолого-педагогического и учебно-методического обеспечения развития информатизации образования Центра информатизации образования Федерального государственного бюджетного научного учреждения «Институт управления образованием Российской академии образования», 105062, г. Москва, ул. Макаренко, д. 5/16, стр. 1Б, polvikpal@mail.ru

Гармаев Баир Заятуевич – кандидат физико-математических наук, научный сотрудник, Институт физического материаловедения Сибирского Отделения РАН, 670000, Россия, республика Бурятия, г. Улан-Удэ, ул. Сахьяновой, 6, каб. 313

Клименко Анна Борисовна – кандидат технических наук, научный сотрудник Научно-исследовательского института многопроцессорных вычислительных систем имени академика А.В. Каляева Южного федерального университета (НИИ МВС ЮФУ), 347935, Россия, Ростовская область, г. Таганрог, ул. 8 Переулок, 15

Лютикова Лариса Адольфовна – кандидат физико-математических наук, заведующая отделом Нейроинформатики и машинного обучения, Институт прикладной математики и автоматизации Кабардино-Балкарского научного центра РАН – филиал Кабардино-Балкарского научного центра РАН (ИПМА КБНЦ РАН), 360000, Россия, Республика Кабардино-Балкария, г. Нальчик, ул. Шортанова, 89а

Мустафаев Арслан Гасанович – доктор технических наук, профессор, Государственное автономное образовательное учреждение высшего образования "Дагестанский государственный университет народного хозяйства", кафедра «Информационные технологии и информационная безопасность», 367015, Россия, Республика Дагестан, г. Махачкала, ул. Атаева, 5, каб. 4.5

Шестаков Александр Валентинович – кандидат технических наук, доцент Южный Федеральный университет, кафедра вычислительной техники, 347902, Россия, Ростовская область, г. Таганрог, ул. Свободы, 24/2

Сидоркина Ирина Геннадьевна – доктор технических наук, профессор, декан факультета Информатики и вычислительной техники Поволжского государственного технологического университета, Йошкар-Ола, Россия E-mail: dekan_fivt@mail.ru

Екатерина Прасолова-Førland - PhD, Норвежский университет науки и технологии (NTNU), Трондхейм, Норвегия E-mail: Ekaterina.Prasolova-Forland@idi.ntnu.no

Голенков Владимир Васильевич – доктор технических наук, профессор, заведующий кафедрой Интеллектуальных информационных технологий Белорусского государственного университета информатики и радиоэлектроники, г. Минск, Республика Беларусь E-mail: golen@bsuir.by

Домошницкий Александр Исаакович – кандидат физико-математических наук, декан естественно-научного факультета Университетского центра в г.Ариэль, Израиль, Самариа E-mail: adom@ariel.ac.il Department of Mathematics and Computer Sciences, The Ariel

University Center of Samaria, 44837 Ariel, ISRAEL

Коробейников Анатолий Григорьевич - доктор технических наук, профессор «Институт земного магнетизма, ионосферы и распространения радиоволн РАН (ИЗМИРАН)», Санкт-Петербургский филиал E-mail: korobeynikov_a_g@mail.ru

Заболеева-Зотова Алла Викторовна, доктор технических наук, профессор Волгоградского технического университета, Волгоград, Россия E-mail: zabzot@gmail.com

Бенкевич Леонид Владимирович - кандидат физических наук и инженерной физики, научный сотрудник Массачусеттского Технологического Института (MIT), обсерватория Хэйстек, Бостон, США E-mail: ibenkev@gmail.com

Морозов Михаил Николаевич - кандидат технических наук, профессор, руководитель лаборатории мультимедиа, заведующий кафедрой Информатики и системного программирования Поволжского государственного технологического университета, Йошкар-Ола, Россия E-mail: mikhail.n.morozov@gmail.com

Олзоева Сэсэг Ивановна - доктор технических наук, профессор, Восточно-Сибирский государственный университет технологий и управления (г. Улан-Уде) E-mail: sseseg@yandex.ru

Курейчик Владимир Викторович - доктор технических наук, профессор, заведующий кафедрой Систем автоматизации проектирования Технологического института «Южного федерального университета» в г.Таганрог, Россия E-mail: ykur@tsure.ru

Филатова Наталья Николаевна - доктор технических наук, профессор, Тверской государственный технический университет, Тверь, Россия E-mail: nfilatova99@mail.ru

Песошин Валерий Андреевич - член-корреспондент Академии наук Республики Татарстан, заслуженный деятель науки Республики Татарстан и Российской Федерации, доктор технических наук, профессор, заслуженный деятель науки и техники Республики Татарстан. Заведующий кафедрой Компьютерных систем Казанского национальный исследовательский университет им. А.Н. Туполева, Казань, Россия E-mail: pesoshin@evm.kstu-kai.ru

Краснов Сергей Викторович - доктор технических наук, профессор, проректор по научно-исследовательской работе, заведующий кафедрой Информатика и системы управления Волжского университета им. Татищева, Тольятти, Россия E-mail: krasnovtlt@mail.ru

Горохов Алексей Витальевич - доктор технических наук, профессор кафедры Прикладной математики и информационных технологий Поволжского государственного технологического университета, Йошкар-Ола, Россия E-mail: agv64@mail.ru

Галанина Наталья Андреевна - доктор технических наук, профессор, Чувашский государственный университет им. И.Н.Ульянова, Чебоксары, Россия E-mail: galaninacheb@mail.ru

Сюзов Владимир Васильевич - доктор технических наук, профессор, заведующий кафедрой Компьютерные системы и сети Московского государственного технического университета им. Н. Э. Баумана, Москва, Россия E-mail: v.suzev@bmstu.ru

Леухин Анатолий Николаевич - доктор физико-математических наук, профессор, заведующий кафедрой Информационной безопасности Поволжского государственного технологического университета, Йошкар-Ола, Россия E-mail: code@volgatech.net

Гвинианидзе Темур Николаевич - Доктор технических наук, профессор,
Государственный университет им. Ак. Церетели Грузия, г. Кутаиси, пр. Тamar-мепе 59.
П.и 4600. temuri1951@mail.ru

Council of Editors

Gelman Viktor Yakovlevich – Doctor of Technical Sciences, Professor, Professor of the Department of Medical Informatics and Physics of the I.I.Mechnikov Northwestern State Medical University, 41 Kirochnaya Str., St. Petersburg, 191015, Russia, gelm@sg2104.spb.edu

Polyakov Viktor Pavlovich – Doctor of Pedagogical Sciences, Professor, Chief Researcher of the Laboratory of Psychological, Pedagogical and Educational methodological support for the development of Informatization of Education of the Center for Informatization of Education of the Federal State Budgetary Scientific Institution "Institute of Education Management of the Russian Academy of Education", 105062, Moscow, Makarenko str., 5/16, p. 1B, polvikpal@mail.ru

Garmaev Bair Zayatuevich – Candidate of Physical and Mathematical Sciences, Researcher, Institute of Physical Materials Science of the Siberian Branch of the Russian Academy of Sciences, 670000, Russia, Republic of Buryatia, Ulan-Ude, Sakhyanova str., 6, room 313

Klimenko Anna Borisovna – Candidate of Technical Sciences, Researcher at the Research Institute of Multiprocessor Computing Systems named after Academician A.V. Kalyaev of the Southern Federal University (Research Institute of the Ministry of Internal Affairs of the Southern Federal University), 347935, Russia, Rostov region, Taganrog, ul. 8 Lane, 15

Lyutikova Larisa Adolfofna – Candidate of Physical and Mathematical Sciences, Head of the Department of Neuroinformatics and Machine Learning, Institute of Applied Mathematics and Automation of the Kabardino-Balkarian Scientific Center of the Russian Academy of Sciences - branch of the Kabardino-Balkarian Scientific Center of the Russian Academy of Sciences (IPMA KBSC RAS), 360000, Russia, Republic of Kabardino-Balkaria, Nalchik, 89a Shortanova str.

Mustafayev Arslan Hasanovich – Doctor of Technical Sciences, Professor, State Autonomous Educational Institution of Higher Education "Dagestan State University of National Economy", Department of "Information Technologies and Information Security", 367015, Russia, Republic of Dagestan, Makhachkala, Ataeva str., 5, office 4.5

Alexander V. Shestakov – Candidate of Technical Sciences, Associate Professor, Southern Federal University, Department of Computer Engineering, 24/2 Svobody str., Taganrog, Rostov Region, 347902, Russia

Sidorkina Irina Gennadievna - Doctor of Technical Sciences, Professor, Dean of the Faculty of Computer Science and Computer Engineering of the Volga State Technological University, Yoshkar-Ola, Russia E-mail: dekan_fivt@mail.ru

Ekaterina Prasolova-Forland - PhD, Norwegian University of Science and Technology (NTNU), Trondheim, Norway E-mail: Ekaterina.Prasolova-Forland@idi.ntnu.no

Golenkov Vladimir Vasilyevich - Doctor of Technical Sciences, Professor, Head of the Department of Intelligent Information Technologies of the Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus E-mail: golen@bsuir.by

Domoshnitsky Alexander Isaakovich - Candidate of Physical and Mathematical Sciences, Dean of the Faculty of Natural Sciences of the University Center in Ariel, Israel, Samaria E-mail: adom@ariel.ac.il Department of Mathematics and Computer Sciences, The Ariel University Center of Samaria, 44837 Ariel, ISRAEL

Korobeynikov Anatoly Grigorievich - Doctor of Technical Sciences, Professor, Institute of

Terrestrial Magnetism, Ionosphere and Radio Wave Propagation of the Russian Academy of Sciences (IZMIRAN), St. Petersburg Branch E-mail: korobeynikov_a_g@mail.ru

Zaboleeva-Zotova Alla Viktorovna, Doctor of Technical Sciences, Professor of Volgograd Technical University, Volgograd, Russia E-mail: zabzot@gmail.com

Leonid V. Benkevich - Candidate of Physical Sciences and Engineering Physics, Researcher at the Massachusetts Institute of Technology (MIT), Haystack Observatory, Boston, USA E-mail: lbenkev@gmail.com

Mikhail N. Morozov - Candidate of Technical Sciences, Professor, Head of the Multimedia Laboratory, Head of the Department of Computer Science and System Programming of the Volga State Technological University, Yoshkar-Ola, Russia E-mail: mikhail.n.morozov@gmail.com

Olzoeva Seseg Ivanovna - Doctor of Technical Sciences, Professor, East Siberian State University of Technology and Management (Ulan-Ude) E-mail: sseseg@yandex.ru

Kureychik Vladimir Viktorovich - Doctor of Technical Sciences, Professor, Head of the Department of Design Automation Systems of the Technological Institute of the Southern Federal University in Taganrog, Russia E-mail: vkur@tsure.ru

Natalia Filatova - Doctor of Technical Sciences, Professor, Tver State Technical University, Tver, Russia E-mail: nfilatova99@mail.ru

Pesoshin Valery Andreevich - Corresponding member of the Academy of Sciences of the Republic of Tatarstan, Honored Scientist of the Republic of Tatarstan and the Russian Federation, Doctor of Technical Sciences, Professor, Honored Worker of Science and Technology of the Republic of Tatarstan. Head of the Department of Computer Systems of Kazan National Research University named after A.N. Tupolev, Kazan, Russia E-mail: pesoshin@evm.kstu-kai.ru

Krasnov Sergey Viktorovich - Doctor of Technical Sciences, Professor, Vice-Rector for Research, Head of the Department of Computer Science and Control Systems of the Volga State University. Tatishcheva, Togliatti, Russia E-mail: krasnovtlt@mail.ru

Gorokhov Alexey Vitalievich - Doctor of Technical Sciences, Professor of the Department of Applied Mathematics and Information Technologies of the Volga State Technological University, Yoshkar-Ola, Russia E-mail: agv64@mail.ru

Galanina Natalia Andreevna - Doctor of Technical Sciences, Professor, I.N.Ulyanov Chuvash State University, Cheboksary, Russia E-mail: galaninacheb@mail.ru

Vladimir V. Syuzev - Doctor of Technical Sciences, Professor, Head of the Department of Computer Systems and Networks of the Bauman Moscow State Technical University, Moscow, Russia E-mail: v.suzev@bmstu.ru

Leukhin Anatoly Nikolaevich - Doctor of Physical and Mathematical Sciences, Professor, Head of the Department of Information Security of the Volga State Technological University, Yoshkar-Ola, Russia E-mail: code@volgatech.net

Gvinianidze Temur Nikolaevich - Doctor of Technical Sciences, Professor, Ak. Tsereteli State University Georgia, Kutaisi, 59 Tamar-mepe Ave., and 4600. temuri1951@mail.ru

Требования к статьям

Журнал является научным. Направляемые в издательство статьи должны соответствовать тематике журнала (с его рубрикаторм можно ознакомиться на сайте издательства), а также требованиям, предъявляемым к научным публикациям.

Рекомендуемый объем от 12000 знаков.

Структура статьи должна соответствовать жанру научно-исследовательской работы. В ее содержании должны обязательно присутствовать и иметь четкие смысловые разграничения такие разделы, как: предмет исследования, методы исследования, апелляция к оппонентам, выводы и научная новизна.

Не приветствуется, когда исследователь, трактуя в статье те или иные научные термины, вступает в заочную дискуссию с авторами учебников, учебных пособий или словарей, которые в узких рамках подобных изданий не могут широко излагать свое научное воззрение и заранее оказываются в проигрышном положении. Будет лучше, если для научной полемики Вы обратитесь к текстам монографий или диссертационных работ оппонентов.

Не превращайте научную статью в публицистическую: не наполняйте ее цитатами из газет и популярных журналов, ссылками на высказывания по телевидению.

Ссылки на научные источники из Интернета допустимы и должны быть соответствующим образом оформлены.

Редакция отвергает материалы, напоминающие реферат. Автору нужно не только продемонстрировать хорошее знание обсуждаемого вопроса, работ ученых, исследовавших его прежде, но и привнести своей публикацией определенную научную новизну.

Не принимаются к публикации избранные части из диссертаций, книг, монографий, поскольку стиль изложения подобных материалов не соответствует журнальному жанру, а также не принимаются материалы, публиковавшиеся ранее в других изданиях.

В случае отправки статьи одновременно в разные издания автор обязан известить об этом редакцию. Если он не сделал этого заблаговременно, рискует репутацией: в дальнейшем его материалы не будут приниматься к рассмотрению.

Уличенные в плагиате попадают в «черный список» издательства и не могут рассчитывать на публикацию. Информация о подобных фактах передается в другие издательства, в ВАК и по месту работы, учебы автора.

Статьи представляются в электронном виде только через сайт издательства <http://www.e-notabene.ru> кнопка "Авторская зона".

Статьи без полной информации об авторе (соавторах) не принимаются к рассмотрению, поэтому автор при регистрации в авторской зоне должен ввести полную и корректную информацию о себе, а при добавлении статьи - о всех своих соавторах.

Не набирайте название статьи прописными (заглавными) буквами, например: «ИСТОРИЯ КУЛЬТУРЫ...» — неправильно, «История культуры...» — правильно.

При добавлении статьи необходимо прикрепить библиографию (минимум 10–15 источников, чем больше, тем лучше).

При добавлении списка использованной литературы, пожалуйста, придерживайтесь следующих стандартов:

- [ГОСТ 7.1-2003 Библиографическая запись. Библиографическое описание. Общие требования и правила составления.](#)
- [ГОСТ 7.0.5-2008 Библиографическая ссылка. Общие требования и правила составления](#)

В каждой ссылке должен быть указан только один диапазон страниц. В теле статьи ссылка на источник из списка литературы должна быть указана в квадратных скобках, например, [1]. Может быть указана ссылка на источник со страницей, например, [1, с. 57], на группу источников, например, [1, 3], [5-7]. Если идет ссылка на один и тот же источник, то в теле статьи нумерация ссылок должна выглядеть так: [1, с. 35]; [2]; [3]; [1, с. 75-78]; [4]....

А в библиографии они должны отображаться так:

[1]

[2]

[3]

[4]....

Постраничные ссылки и сноски запрещены. Если вы используете сноску, не содержащую ссылку на источник, например, разъяснение термина, включите сноску в текст статьи.

После процедуры регистрации необходимо прикрепить аннотацию на русском языке, которая должна состоять из трех разделов: Предмет исследования; Метод, методология исследования; Новизна исследования, выводы.

Прикрепить 10 ключевых слов.

Прикрепить саму статью.

Требования к оформлению текста:

- Кавычки даются уголками (« ») и только кавычки в кавычках — лапками (" ").
- Тире между датами дается короткое (Ctrl и минус) и без отбивок.
- Тире во всех остальных случаях дается длинное (Ctrl, Alt и минус).
- Даты в скобках даются без г.: (1932–1933).
- Даты в тексте даются так: 1920 г., 1920-е гг., 1540–1550-е гг.
- Недопустимо: 60-е гг., двадцатые годы двадцатого столетия, двадцатые годы XX столетия, 20-е годы XX столетия.
- Века, король такой-то и т.п. даются римскими цифрами: XIX в., Генрих IV.
- Инициалы и сокращения даются с пробелом: т. е., т. д., М. Н. Иванов. Неправильно: М.Н. Иванов, М.Н. Иванов.

ВСЕ СТАТЬИ ПУБЛИКУЮТСЯ В АВТОРСКОЙ РЕДАКЦИИ.

По вопросам публикации и финансовым вопросам обращайтесь к администратору Зубковой Светлане Вадимовне

E-mail: info@nbpublish.com

или по телефону +7 (966) 020-34-36

Подробные требования к написанию аннотаций:

Аннотация в периодическом издании является источником информации о содержании статьи и изложенных в ней результатах исследований.

Аннотация выполняет следующие функции: дает возможность установить основное

содержание документа, определить его релевантность и решить, следует ли обращаться к полному тексту документа; используется в информационных, в том числе автоматизированных, системах для поиска документов и информации.

Аннотация к статье должна быть:

- информативной (не содержать общих слов);
- оригинальной;
- содержательной (отражать основное содержание статьи и результаты исследований);
- структурированной (следовать логике описания результатов в статье);

Аннотация включает следующие аспекты содержания статьи:

- предмет, цель работы;
- метод или методологию проведения работы;
- результаты работы;
- область применения результатов; новизна;
- выводы.

Результаты работы описывают предельно точно и информативно. Приводятся основные теоретические и экспериментальные результаты, фактические данные, обнаруженные взаимосвязи и закономерности. При этом отдается предпочтение новым результатам и данным долгосрочного значения, важным открытиям, выводам, которые опровергают существующие теории, а также данным, которые, по мнению автора, имеют практическое значение.

Выводы могут сопровождаться рекомендациями, оценками, предложениями, гипотезами, описанными в статье.

Сведения, содержащиеся в заглавии статьи, не должны повторяться в тексте аннотации. Следует избегать лишних вводных фраз (например, «автор статьи рассматривает...», «в статье рассматривается...»).

Исторические справки, если они не составляют основное содержание документа, описание ранее опубликованных работ и общеизвестные положения в аннотации не приводятся.

В тексте аннотации следует употреблять синтаксические конструкции, свойственные языку научных и технических документов, избегать сложных грамматических конструкций.

Гонорары за статьи в научных журналах не начисляются.

Цитирование или воспроизведение текста, созданного ChatGPT, в вашей статье

Если вы использовали ChatGPT или другие инструменты искусственного интеллекта в своем исследовании, опишите, как вы использовали этот инструмент, в разделе «Метод» или в аналогичном разделе вашей статьи. Для обзоров литературы или других видов эссе, ответов или рефератов вы можете описать, как вы использовали этот инструмент, во введении. В своем тексте предоставьте prompt - командный вопрос, который вы использовали, а затем любую часть соответствующего текста, который был создан в ответ.

К сожалению, результаты «чата» ChatGPT не могут быть получены другими читателями, и хотя невозстановимые данные или цитаты в статьях APA Style обычно цитируются как личные сообщения, текст, сгенерированный ChatGPT, не является сообщением от человека.

Таким образом, цитирование текста ChatGPT из сеанса чата больше похоже на совместное использование результатов алгоритма; таким образом, сделайте ссылку на автора алгоритма записи в списке литературы и приведите соответствующую цитату в тексте.

Пример:

На вопрос «Является ли деление правого полушария левого полушария реальным или метафорой?» текст, сгенерированный ChatGPT, показал, что, хотя два полушария мозга в некоторой степени специализированы, «обозначение, что люди могут быть охарактеризованы как «левополушарные» или «правополушарные», считается чрезмерным упрощением и популярным мифом» (OpenAI, 2023).

Ссылка в списке литературы

OpenAI. (2023). ChatGPT (версия от 14 марта) [большая языковая модель].
<https://chat.openai.com/chat>

Вы также можете поместить полный текст длинных ответов от ChatGPT в приложение к своей статье или в дополнительные онлайн-материалы, чтобы читатели имели доступ к точному тексту, который был сгенерирован. Особенно важно задокументировать точный созданный текст, потому что ChatGPT будет генерировать уникальный ответ в каждом сеансе чата, даже если будет предоставлен один и тот же командный вопрос. Если вы создаете приложения или дополнительные материалы, помните, что каждое из них должно быть упомянуто по крайней мере один раз в тексте вашей статьи в стиле APA.

Пример:

При получении дополнительной подсказки «Какое представление является более точным?» в тексте, сгенерированном ChatGPT, указано, что «разные области мозга работают вместе, чтобы поддерживать различные когнитивные процессы» и «функциональная специализация разных областей может меняться в зависимости от опыта и факторов окружающей среды» (OpenAI, 2023; см. Приложение А для полной расшифровки). .

Ссылка в списке литературы

OpenAI. (2023). ChatGPT (версия от 14 марта) [большая языковая модель].
<https://chat.openai.com/chat> Создание ссылки на ChatGPT или другие модели и программное обеспечение ИИ

Приведенные выше цитаты и ссылки в тексте адаптированы из шаблона ссылок на программное обеспечение в разделе 10.10 Руководства по публикациям (Американская психологическая ассоциация, 2020 г., глава 10). Хотя здесь мы фокусируемся на ChatGPT, поскольку эти рекомендации основаны на шаблоне программного обеспечения, их можно адаптировать для учета использования других больших языковых моделей (например, Bard), алгоритмов и аналогичного программного обеспечения.

Ссылки и цитаты в тексте для ChatGPT форматируются следующим образом:

OpenAI. (2023). ChatGPT (версия от 14 марта) [большая языковая модель].
<https://chat.openai.com/chat>

Цитата в скобках: (OpenAI, 2023)

Описательная цитата: OpenAI (2023)

Давайте разберем эту ссылку и посмотрим на четыре элемента (автор, дата, название и

источник):

Автор: Автор модели OpenAI.

Дата: Дата — это год версии, которую вы использовали. Следуя шаблону из Раздела 10.10, вам нужно указать только год, а не точную дату. Номер версии предоставляет конкретную информацию о дате, которая может понадобиться читателю.

Заголовок. Название модели — «ChatGPT», поэтому оно служит заголовком и выделено курсивом в ссылке, как показано в шаблоне. Хотя OpenAI маркирует уникальные итерации (например, ChatGPT-3, ChatGPT-4), они используют «ChatGPT» в качестве общего названия модели, а обновления обозначаются номерами версий.

Номер версии указан после названия в круглых скобках. Формат номера версии в справочниках ChatGPT включает дату, поскольку именно так OpenAI маркирует версии. Различные большие языковые модели или программное обеспечение могут использовать различную нумерацию версий; используйте номер версии в формате, предоставленном автором или издателем, который может представлять собой систему нумерации (например, Версия 2.0) или другие методы.

Текст в квадратных скобках используется в ссылках для дополнительных описаний, когда они необходимы, чтобы помочь читателю понять, что цитируется. Ссылки на ряд общих источников, таких как журнальные статьи и книги, не включают описания в квадратных скобках, но часто включают в себя вещи, не входящие в типичную рецензируемую систему. В случае ссылки на ChatGPT укажите дескриптор «Большая языковая модель» в квадратных скобках. OpenAI описывает ChatGPT-4 как «большую мультимодальную модель», поэтому вместо этого может быть предоставлено это описание, если вы используете ChatGPT-4. Для более поздних версий и программного обеспечения или моделей других компаний могут потребоваться другие описания в зависимости от того, как издатели описывают модель. Цель текста в квадратных скобках — кратко описать тип модели вашему читателю.

Источник: если имя издателя и имя автора совпадают, не повторяйте имя издателя в исходном элементе ссылки и переходите непосредственно к URL-адресу. Это относится к ChatGPT. URL-адрес ChatGPT: <https://chat.openai.com/chat>. Для других моделей или продуктов, для которых вы можете создать ссылку, используйте URL-адрес, который ведет как можно более напрямую к источнику (т. е. к странице, на которой вы можете получить доступ к модели, а не к домашней странице издателя).

Другие вопросы о цитировании ChatGPT

Вы могли заметить, с какой уверенностью ChatGPT описал идеи латерализации мозга и то, как работает мозг, не ссылаясь ни на какие источники. Я попросил список источников, подтверждающих эти утверждения, и ChatGPT предоставил пять ссылок, четыре из которых мне удалось найти в Интернете. Пятая, похоже, не настоящая статья; идентификатор цифрового объекта, указанный для этой ссылки, принадлежит другой статье, и мне не удалось найти ни одной статьи с указанием авторов, даты, названия и сведений об источнике, предоставленных ChatGPT. Авторам, использующим ChatGPT или аналогичные инструменты искусственного интеллекта для исследований, следует подумать о том, чтобы сделать эту проверку первоисточников стандартным процессом. Если источники являются реальными, точными и актуальными, может быть лучше прочитать эти первоисточники, чтобы извлечь уроки из этого исследования, и перефразировать или процитировать эти статьи, если применимо, чем использовать их интерпретацию модели.

Материалы журналов включены:

- в систему Российского индекса научного цитирования;
- отображаются в крупнейшей международной базе данных периодических изданий Ulrich's Periodicals Directory, что гарантирует значительное увеличение цитируемости;
- Всем статьям присваивается уникальный идентификационный номер Международного регистрационного агентства DOI Registration Agency. Мы формируем и присваиваем всем статьям и книгам, в печатном, либо электронном виде, оригинальный цифровой код. Префикс и суффикс, будучи прописанными вместе, образуют определяемый, цитируемый и индексируемый в поисковых системах, цифровой идентификатор объекта — digital object identifier (DOI).

[Отправить статью в редакцию](#)

Этапы рассмотрения научной статьи в издательстве NOTA BENE.



Содержание

Притыкин Ф.Н. Использование компьютерного моделирования движения руки антропоморфного робота для определения положения его основания относительно объектов манипулирования	1
Макаров И.С., Ларин Д.В., Воробьева Е.Г., Емелин Д.П., Карташов Д.А. Влияние асинхронных и многопоточных моделей обработки запросов на производительность серверных веб-приложений	13
Золотухина Д.Ю. Сравнительный анализ индексационных стратегий в PostgreSQL при различных сценариях нагрузки	21
Макаров К.С., Фаткин Р.И. Скриншотное тестирование как многоаспектный вид автоматизированной динамической верификации веб-приложений	32
Сергеев Д.А., Родионов Д.Г., Поляков П.А., Голиков Г.И., Старченкова О.Д., Дмитриев Н.Д., Конников Е.А. Интеллектуальная система мониторинга и адаптации маршрута беспилотных летательных аппаратов на основе нейросетевого анализа объектов риска	55
Савельева Ю.О., Ильин А.В., Лукьянов А.А., Милихина К.М. Разработка прикладного приложения для автоматизации проектирования в КОМПАС-3D: таблица контроля качества основного металла	71
Плетнева А.О., Янчус В.Э. Разработка методики тестирования мобильного приложения на основе технологии ай-трекинга	92
Англоязычные метаданные	106

Contents

Pritykin F.N. Using computer simulation of the hand movement of an anthropomorphic robot to determine the position of its base relative to objects of manipulation	1
Makarov I.S., Larin D.V., Vorobeva E.G., Emelin D.P., Kartashov D.A. The impact of asynchronous and multithreaded query processing models on the performance of server-side web applications	13
Zolotukhina D. Comparative analysis of indexing strategies in PostgreSQL under various load scenarios	21
Makarov K.S., Fatkin R.I. Screenshot testing as a multi-aspect type of automated dynamic verification for web applications	32
Sergeev D.A., Rodionov D.G., Polyakov P.A., Golikov G.I., Starchenkova O.D., Dmitriev N.D., Konnikov E.A. Intelligent Monitoring and Route Adaptation System for Drones based on Neural Network Risk Analysis	55
Savelieva Y.O., Il'in A.V., Luk'yanov A.A., Milihina K.M. Development of an Application for Design Automation in KOMPAS-3D: Quality Control Table for Base Metal	71
Pletneva A.O., Yanchus V.E. Development of a methodology for testing a mobile application based on eye-tracking technology	92
Metadata in english	106

Программные системы и вычислительные методы

Правильная ссылка на статью:

Притыкин Ф.Н. Использование компьютерного моделирования движения руки антропоморфного робота для определения положения его основания относительно объектов манипулирования // Программные системы и вычислительные методы. 2025. № 1. DOI: 10.7256/2454-0714.2025.1.70232 EDN: UUCVZP URL: https://nbpublish.com/library_read_article.php?id=70232

Использование компьютерного моделирования движения руки антропоморфного робота для определения положения его основания относительно объектов манипулирования

Притыкин Федор Николаевич

доктор технических наук

профессор, Омский государственный технический университет

644050, Россия, Омская область, г. Омск, проспект Мира, 11, ауд. 3в-516

✉ pritykin@mail.ru



[Статья из рубрики "Операционные системы"](#)

DOI:

10.7256/2454-0714.2025.1.70232

EDN:

UUCVZP

Дата направления статьи в редакцию:

25-03-2024

Аннотация: Предметом исследования является использование интеллектуальной системы управления движением руки антропоморфного робота при выполнении операций размещения объектов манипулирования в контейнер заданных размеров. Объектом исследования является процесс определения параметров относительного положения основания руки по отношению к объектам манипулирования и контейнера, при которых возможно выполнение двигательных заданий. Автор подробно рассматривает алгоритм определения положения основания руки антропоморфного робота с использованием синтеза движений по вектору скоростей при решении задачи установки объектов манипулирования заданных, в виде прямоугольных призм в контейнер. Особое внимание в статье уделяется методике определения центра системы координат, связанного с основанием робота в неподвижном пространстве и вычисления целевых точек, в которые перемещается центр выходного звена при различных

положениях ранее установленных в контейнер объектов манипулирования. Суть метода состоит в использовании компьютерного моделирования движения антропоморфного робота с использованием синтеза движений по вектору скоростей с оценкой взаимного положения механизма руки и запретных зон. В качестве запретных зон выступают установленные ранее объекты манипулирования располагающихся внутри контейнера и боковые стенки самого контейнера. Основными выводами представленного исследования является возможность использования разработанного алгоритма для проверки синтеза движений руки при заданных геометрических параметрах, задающих положение антропоморфного робота, конвейера и контейнера, при которых отсутствует возникновение тупиковых ситуаций. Новизна исследования состоит в разработке метода, основанного на итерационном поиске значений параметров взаимного положения основания руки, конвейера и контейнера на каждой итерации при возникновении тупиковых ситуаций. Представлены результаты расчетов положения основания руки робота и промежуточных конфигураций, построенных с использованием компьютерного моделирования движений на основе использования разработанного алгоритма. Проведенные исследования могут быть использованы при разработке информационно-управляющих комплексов подвижных объектов, в частности при разработке интеллектуальных систем управления автономно функционирующих антропоморфных роботов в организованных средах.

Ключевые слова:

интеллектуальные системы управления, механизмы манипуляторов, геометрическое моделирование, компьютерное моделирование, синтез движений, антропоморфные роботы, тупиковые ситуации, модель кинематической цепи, объекты манипулирования, выходное звено

1. Введение

Анализ современных исследований связанных с внедрением роботов в различных сферах деятельности человека в развитых странах мира показывает, что среднегодовые темпы роста производства роботов и их использования за период 2019 по 2024 годы составляет от 10 до 12 процентов. Одно из направлений где могут быть использованы роботы является разработка так называемых "умных складов". Как известно построение любого "умного склада" основано на использовании двух глобальных компонентов это информатизация и робототизация. Руководители в сфере логистики ожидают к 2024 году достижения уровня использования робототехники для управления входящими запасами до 24%, для упаковки исходящих грузов - 22% и для приема грузов до 20%. При автоматизации работ в складских помещениях могут быть использованы антропоморфные роботы оснащенные системами управления движением с использованием элементов искусственного интеллекта. Поэтому разработка интеллектуальных систем управления движением механизмами рук антропоморфных роботов способных заменить человека при выполнении указанных работ, где необходима оценка рабочей сцены и изменения в соответствии с этим поведения робота в организованной среде является актуальной [\[1-4\]](#). Решение указанной проблемы связано с представлением знаний, накопленных роботом в процессе его функционирования [\[5-8\]](#). Интеллектуальное управление предполагает использование виртуальных моделей на начальном этапе планирования поведения робота при определении достижимости тех или иных целевых точек в сложноорганизованной среде [\[9-12\]](#).

При компьютерном моделировании движений роботов применяют метод, основанный на использовании матриц частных передаточных отношений [13-15]. С помощью данных матриц устанавливается взаимосвязь скоростей выходного звена (ВЗ) и обобщенных скоростей. При использовании указанного метода часто необходимо определять взаимное положение антропоморфного робота, при котором его рука способна достигать заданные целевые точки с учетом изменяющейся с течением времени окружающей среды. Анализ источников показывает, что вопросы возникновения тупиковых ситуаций при компьютерном управлении движением антропоморфных роботов в организованных средах не достаточно исследованы. Поэтому существует необходимость в разработке алгоритма позволяющего заранее исключить возникновение тупиковых ситуаций при компьютерном управлении движением антропоморфного робота и объектов манипулирования [16,17].

2. Постановка задачи

Рассмотрим алгоритм определения положения основания антропоморфного робота с использованием компьютерного моделирования движений на примере разработки так называемого "умного склада", где необходимо размещать объекты манипулирования, имеющие форму прямоугольных параллелепипедов в контейнер заданных размеров. На рисунке 1 представлены объекты манипулирования 3 различных размеров, которые перемещаются в произвольном порядке по конвейеру 4. Данные объекты необходимо размещать в контейнере 5 с использованием руки антропоморфного робота 2. На рисунке 1 приведено условное изображение руки антропоморфного робота с обозначением осей и поворотов во вращательных кинематических парах, задающих значения обобщенных координат $q_2 - q_8$. Рука антропоморфного робота также может смещаться поступательно вдоль направляющей 1 за счет изменения обобщенной координаты q_1 . Кинематическая схема этого механизма изображена на рисунке 2. Изображение общего вида антропоморфного робота на рисунке 2 заимствовано с сайта НПО «Андроидная техника».

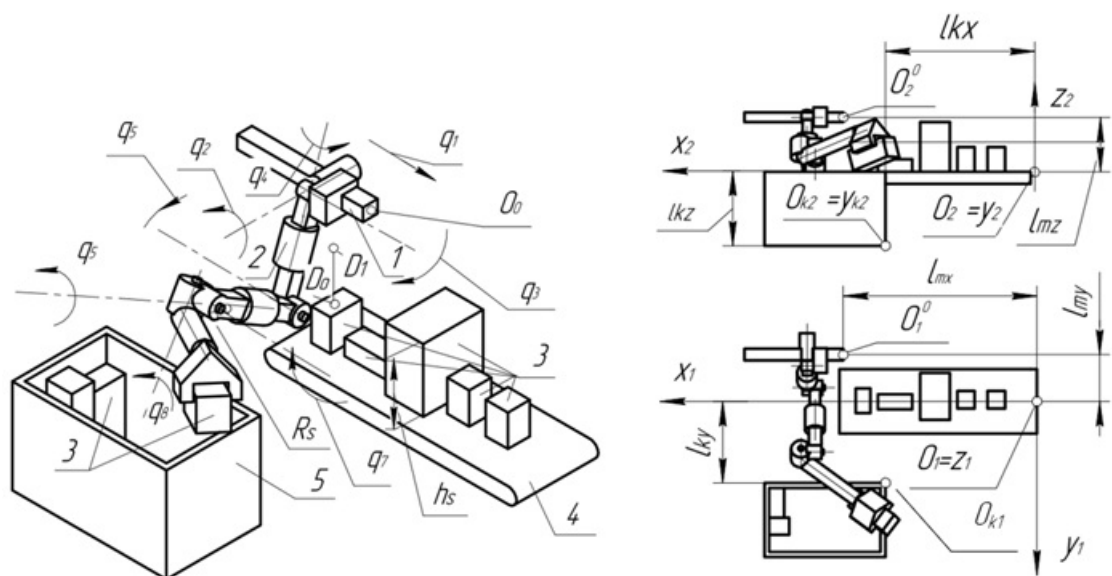


Рис. 1 Положение объектов манипулирования, контейнера, конвейера и руки антропоморфного робота

При автоматизированном управлении движением механизма руки антропоморфного робота в сложно организованном пространстве необходимо обеспечивать достижимость

захватом заданных объектов манипулирования на конвейере и установку их в контейнер. Заметим, что боковые стенки контейнера и установленные в контейнере объекты манипулирования (прямоугольные призмы) выступают в качестве запретных зон при компьютерном управлении движением механизма руки (см. рис. 1). Положение запретных зон (объектов манипулирования) в решаемой задаче может изменяться с течением времени. На рисунке 1 положение основания руки робота определяется точкой $O^0(O_1^0, O_2^0)$. При большом числе степеней подвижности руки антропоморфного робота, а также при наличии запретных зон во внешней среде и ограничений на предельные значения обобщенных координат задача определения достижимости совокупности целевых точек при заданном положении основания (точки $O^0(O_1^0, O_2^0)$) аналитическими методами не может быть решена. Поэтому при решении указанной выше задачи необходимо использовать алгоритм, в основу которого положено последовательное компьютерное моделирование движений руки с оценкой возможности достижения заданных целевых точек при различном положении основания руки, контейнера и различном положении объектов манипулирования внутри контейнера. Ранее подобная задача, связанная с определением положения робота относительно запретных зон, решалась в работе [18].

Если возникают тупиковые ситуации при автоматизированном синтезе движений по вектору скоростей, то необходимо изменять взаимное положение контейнера и основания руки антропоморфного робота по отношению к конвейеру. Тупиковая ситуация возникает тогда, когда алгоритм синтеза движений не способен вычислить следующую промежуточную конфигурацию при смещении центра ВЗ в следующую точку заданной траектории [14].

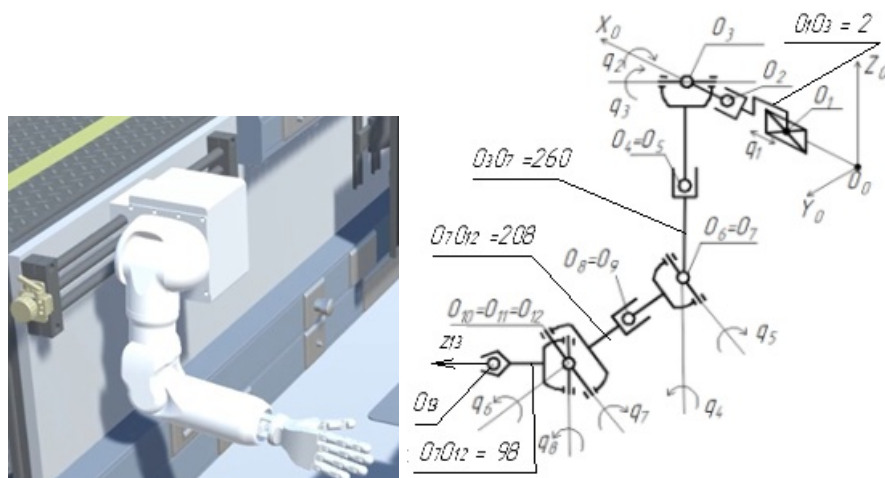


Рис. 2. Общий вид и кинематическая схема руки антропоморфного робота

$$nm = 13, n = 8 \text{ (M4-1-2-12-3-12-1-12-3-12-1-2-12)}$$

После возникновения тупиковой ситуации компьютерное моделирование движения осуществляется вновь от начальных до совокупностей целевых точек, но при других значениях параметров $l_{kx}, l_{ky}, l_{kz}, l_{mx}, l_{my}$ и l_{mz} характеризующих взаимное положение контейнера и основания механизма руки робота относительно конвейера (см. рис. 1).

3. Теория

Рассмотрим кинематическую схему антропоморфного робота, представленную на рисунке 2. В соответствии с методикой обозначения структурных моделей открытых

кинематических цепей, принятой в работе [14] данный механизм, имеет обозначение М4-1-2-12-3-12-1-12-3-12-1-2-12. На указанном рисунке узловые точки задающие начала систем координат, связанных со звеньями механизма обозначены точками O_1, O_2, \dots, O_{nm} . Где параметр nm определяет число систем координат, используемых при задании модели кинематической цепи [14]. Расстояния между узловыми точками на рисунке заданы длинами $O_1O_3 = 2, O_3O_7 = 260$, и т.п. в миллиметрах. Заметим, что некоторые узловые точки могут совпадать между собой. Известно, что совокупность мгновенных состояний, удовлетворяющих заданной точности позиционирования d центра ВЗ, определяет область Ω^Q принадлежащая Γ^Q в многомерном пространстве обобщенных скоростей Q . Где, Γ^Q обозначена p -плоскость заданная линейной системой уравнений задающей зависимость скоростей простейших движений ВЗ от обобщенных скоростей q_i (верхний индекс обозначений Q означает задание и исследование геометрических объектов Ω^Q и Γ^Q в многомерном пространстве Q [19-20]).

Решение задачи определения значений параметров $l_{kx}, l_{ky}, l_{kz}, l_{mx}, l_{my}$ и l_{mz} при которых обеспечена установка объектов манипулирования в контейнер состоит из следующих этапов:

1. Выбирают конечную совокупность точек-претендентов O^0 и точек O_k соответственно задающих положения центров систем координат, связанных с основанием руки робота и базовой точки контейнера. Принимаются максимальные и минимальные значения параметров $l_{kx}^{max}, l_{ky}^{max}, l_{kz}^{max}, l_{mx}^{max}, l_{my}^{max}, l_{mz}^{max}, l_{kx}^{min}, l_{ky}^{min}, l_{kz}^{min}, l_{mx}^{min}, l_{my}^{min}$ и l_{mz}^{min} .
2. Задаются положения и форма геометрических объектов, характеризующих окружающую внешнюю среду. Для рассматриваемого примера необходимо задать совокупность многогранников задающих грани боковых стенок контейнера, положение которого определяется параметрами l_{kx}, l_{ky} и l_{kz} в неподвижной системе координат $Oxuz$. Также необходимо задать в подвижных системах координат, неподвижно связанных с объектами манипулирования (четырёхугольными прямоугольными призмами) положение граней, определяющих их размеры и форму.
3. Выбирается совокупность начальных точек на конвейере, и конечных положений целевых точек внутри контейнера (эти точки задают конечные положения центра ВЗ руки антропоморфного робота). Заметим, что объекты манипулирования могут перемещаться конвейером в различном порядке. Положения двух объектов манипулирования 3^1 на одном из этапов компьютерного моделирования изображены на горизонтальной проекции рисунка 3.

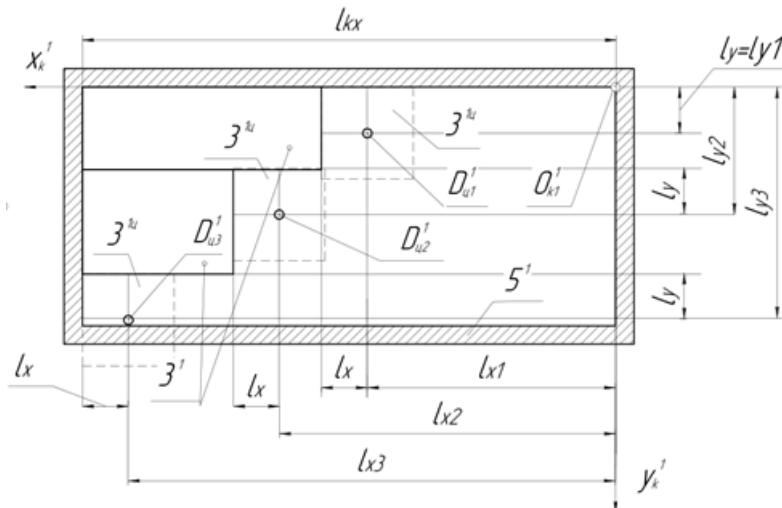


Рис. 3. Вычисление положения целевых точек $D_{цi}$ на горизонтальной проекции при различном положении объектов манипулирования в контейнере

4. Задаются промежуточные целевые точки (D_1, \dots т.п.) определяющие траектории движения центра ВЗ состоящие из отрезков прямых.

5. Организуется цикл построения и анализа положений промежуточных конфигураций и объектов манипулирования относительно запретных зон. При этом используется компьютерное моделирование движения центра ВЗ и объектов манипулирования до конечных целевых точек по критерию минимизации объема движения, если пересечения механизма манипулятора и перемещаемого объекта манипулирования с запретными зонами отсутствует. Если возникают указанные пересечения с запретными зонами исследуются другие допустимые мгновенные состояния, определяемые вектором приращений обобщенных координат [14]. Если при компьютерном моделировании возникают тупиковые ситуации набор параметров $l_{kx}, l_{ky}, l_{kz}, l_{mx}, l_{my}$ и l_{mz} необходимо изменять. Заметим, что параметры l_{mx} и l_{kx} необходимо исследовать с учетом максимального значения обобщенной координаты q_1 .

На рисунке 3 представлена горизонтальная проекция контейнера, в котором необходимо определять положения объектов манипулирования. На рисунке изображены два ранее установленные объекты манипулирования 3^1 . Определение положений целевых точек $D_{ц1}, D_{ц2}$ и $D_{ц3}$ необходимо осуществлять с учетом параметров l_x и l_y задающих размеры перемещаемого объекта манипулирования. После вычисления положения указанных точек определяется пересечения объекта $3^{1ц}$ с гранями контейнера. Как видно из рисунка данному условию не удовлетворяет точка $D_{ц3}$. Из точек $D_{ц1}$ и $D_{ц2}$ выбирается точка, которая имеет какое либо наименьшее значение l_{yi} или наибольшее l_{xi} . Если разность координат $l_{x1} - l_{x2} > l_x$ то выбирается точка у которой будет большее значение l_{yi} . Где параметры l_{xi} (l_{x1}, l_{x2}, l_{x3}) и l_{yi} (l_{y1}, l_{y2}, l_{y3}) определяют положения целевых точек- претендентов перемещаемого объекта манипулирования на горизонтальной проекции. На рисунке 1 точка D_0 задает точку, в которую смещается центр ВЗ при захвате объекта манипулирования. После захвата объекта манипулирования центр ВЗ из точки D_0 перемещается в точку D_1 и после этого к целевым точкам $D_{ц1}$ и т.п.

Алгоритм определения положения контейнера и основания руки антропоморфного робота относительно конвейера представлен на рисунке 4.

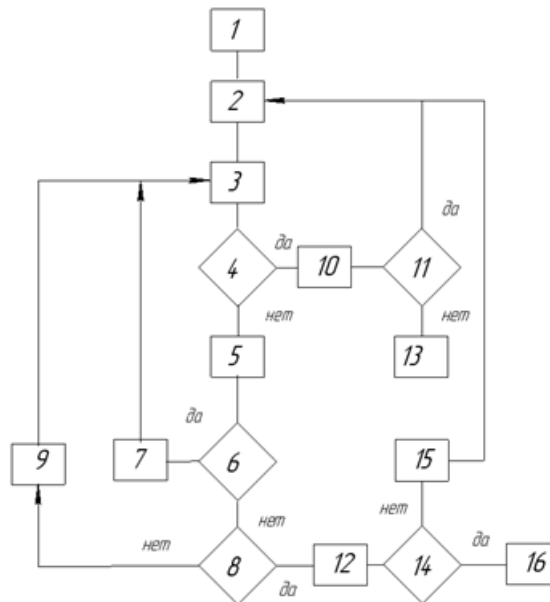


Рис. 4 — Алгоритм вычисления основания механизма руки антропоморфного робота

На рисунке 4 приняты следующие обозначения: 1 – задание начальных значений параметров l_{kx} , l_{ky} , l_{kz} и l_{mx} , l_{my} и l_{mz} , приращений DI_{kx} , DI_{ky} , DI_{kz} и DI_{mx} , DI_{my} и DI_{mz} и предельных значений l_{kx}^{max} , l_{ky}^{max} , l_{kz}^{max} , l_{mx}^{max} , l_{my}^{max} , l_{mz}^{max} , l_{kx}^{min} , l_{ky}^{min} , l_{kz}^{min} , l_{mx}^{min} , l_{my}^{min} и l_{mz}^{min} . Задание информации о начальном положении механизма руки и модели кинематической цепи (q_1 , q_n , nm , n_{kod} , l_i , r , n , где n_{kod} – коды преобразований систем координат [14]), l_i – длины вензев механизма, r – размерность вектора простейших движений ВЗ, n – число обобщенных координат), задание предельных значений обобщенных координат, задание максимальных значений параметров k_i^{max} , k_i^{max} определяющих максимальные значения задающие размеры области Ω^Q ; 2 – вычисление последовательности целевых точек-претендентов конечных положений центра ВЗ (D_1 , $D_{ц1}$, $D_{ц2}$ см. рис. 1, рис. 3); 3 – вычисление вектора обобщенных скоростей \mathbf{Q} обеспечивающего направление движения центра ВЗ к целевым точкам (D_1 , $D_{ц1}$, $D_{ц2}$ и т.п.); 4 – все значения вектора \mathbf{Q} использованы $k_i > k_i^{max}$ (возникает тупиковая ситуация ?); 5 – реализация значений вектора \mathbf{Q} и вычисление следующей конфигурации ($q_1 = q_1 + Dq_1$, ..., $q_n = q_n + Dq_n$, при этом принято допущение равенства значений обобщенных скоростей и приращений $\dot{q}_1 = Dq_1$, $\dot{q}_2 = Dq_2$, ..., $\dot{q}_n = Dq_n$); 6 – расчетная конфигурация пересекает запретные зоны, или не удовлетворяет предельным значениям обобщенных координат; 7 – изменение значений параметров k_i ($k_i = k_i + 1$) задающих координаты точек внутри области Ω^Q ; 8 – целевая точка $D_{ц}$ достигнута; 9 – $k_i = 0$, вычисление матрицы частных передаточных отношений и вектора простейших движений ВЗ; 10 – изменение значений параметров l_{kx} , l_{ky} , l_{kz} и l_{mx} , l_{my} и l_{mz} ($l_{kx} + DI_{kx}$ и т.п.), 11 – значения параметров удовлетворяют заданным предельным значениям $l_{kx}^{min} < l_{kx} < l_{kx}^{max}$, $l_{ky}^{min} < l_{ky} < l_{ky}^{max}$, $l_{kz}^{min} < l_{kz} < l_{kz}^{max}$, $l_{mx}^{min} < l_{mx} < l_{mx}^{max}$, $l_{my}^{min} < l_{my} < l_{my}^{max}$, $l_{mz}^{min} < l_{mz} < l_{mz}^{max}$; 12 – заданная целевая точка может быть достигнута, переход с текущими значениями параметров l_{kx} ,

l_{ky} , l_{kz} и l_{mx} , l_{my} и l_{mz} к синтезу движений к другим целевым точкам; 13 – при заданных размерах контейнера и объектов манипулирования целевые точки не могут быть достигнуты; 14 – все целевые точки достигнуты; 15 – изменение последовательности размещения объектов манипулирования и переход к синтезу движений к другим целевым точкам; 16 – значения параметров l_{kx} , l_{ky} , l_{kz} и l_{mx} , l_{my} и l_{mz} позволяют установку объектов манипулирования в контейнер.

4. Результаты экспериментов

На рисунке 5 на фронтальной и профильной плоскостях проекций представлен фрагмент синтеза движений руки антропоморфного робота кинематическая схема которого представлена на рисунке 2. На рисунке изображения проекций объектов манипулирования обозначены 3^2 и 3^3 . Соответственно изображения проекций точек O_1 , $D_{ц}$ и контейнера 5 обозначены O_1^2 , O_1^3 , $D_{ц}^2$, $D_{ц}^3$, 5^2 , и 5^3 . При решении тестовой задачи были приняты следующие размеры геометрических объектов – размеры контейнера (соответственно длина, ширина и высота в миллиметрах) 500x325x200, размеры объектов манипулирования 210x110x85 - 3 шт. и 110x50x50 - 3 шт. Компьютерное моделирование указало на то, что при значениях параметров $l_{kx} = 0$, $l_{ky} = -140$, $l_{kz} = -200$, $l_{mx} = -50$, $l_{my} = -380$ и $l_{mz} = 330$ рука антропоморфного робота может обеспечить установку заданных объектов манипулирования внутри контейнера.

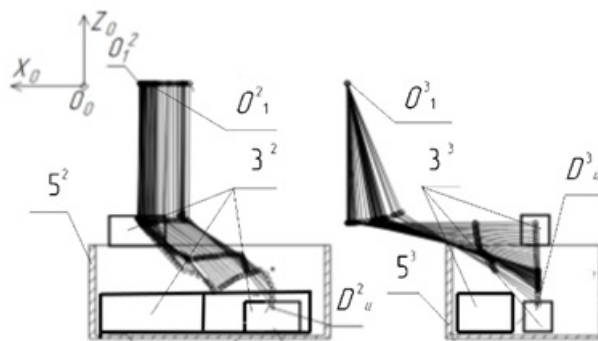


Рис. 5 - Результат фрагмента компьютерного моделирования движения руки антропоморфного робота на фронтальной и профильной плоскостях проекций при установке объекта манипулирования в контейнер

Выводы и заключение

Как правило различные изделия поставляются в различных упаковках, которые могут иметь форму прямоугольных параллелепипедов различных размеров. Хранение, поиск, систематизация и транспортировка указанных изделий в складских помещениях требует больших временных затрат обслуживающего персонала. Использование "умных складов", где выполняются процессы погрузки и разгрузки на стелаж и со стелаж и в транспортные контейнеры антропоморфными роботами при транспортировке этих изделий к местам их хранения может в некоторой степени автоматизировать процесс обработки грузов. В конечном итоге это позволяет обеспечить увеличение производительности и сокращения времени транспортировки изделий в складских помещениях. Результаты компьютерного моделирования движения антропоморфного робота с использованием разработанного алгоритма показали возможность предвидеть и преодолевать тупиковые ситуации при компьютерном управлении движением с помощью изменения параметров взаимного расположения руки антропоморфного робота относительно объектов окружающей среды. Проведенные исследования могут быть использованы при

разработке информационно-управляющих комплексов подвижных объектов, в частности при разработке систем управления автономно функционирующих антропоморфных роботов в складских помещениях.

Библиография

1. Щербатов И. А. Интеллектуальное управление робототехническими системами в условиях неопределённости // Вестник АГТУ, сер. Управление, вычислительная техника и информатика. 2010. № 1. С. 73-77.
2. Ющенко А. С. Интеллектуальное планирование в деятельности роботов // Мехатроника, автоматизация, управление. 2005. № 3. С. 5-18.
3. Афонин В.Л., Базров Б.М., Слепцов В.В. Тензорное исчисление в задачах управления и представления знаний в интеллектуальных обрабатывающих технологических системах // Проблемы машиностроения и автоматизации. 2021. № 1. С. 9-18.
4. Мелехин В.Б., Хачумов М.В. Планирование целенаправленной деятельности автономным интеллектуальным роботом с обновлением знаний в кратковременной памяти // Мехатроника, автоматизация, управление. 2024. Т. 25. № 2. С. 79-92.
5. Мелехин В.Б., Хачумов М.В. Когнитивные инструменты абстрактного мышления автономных интеллектуальных мобильных систем // Мехатроника, автоматизация, управление. 2023. Т. 24, № 6. С. 317-326.
6. Манько С.В., Лохин В.М., Крайнов Н.В., Малько А.Н. Алгоритмы интеллектуального управления реконфигурируемыми роботами в компоновке колеса и многоагентными системами на их основе // Мехатроника, автоматизация, управление. 2022. Т. 23, № 8. С. 420-429.
7. Дубенко Ю. В., Дышкант Е. Е., Тимченко Н. Н., Рудешко Н. А. Гибридный алгоритм формирования кратчайшей траектории, основанный на применении многоагентного обучения с подкреплением и обмена опытом // Вестник компьютерных и информационных технологий. 2021. № 11. С. 13-26.
8. Kutlubaev, I.M. Bogdanov, A.A., Novoseltsev, N.V., Krasnobaev, M.V., Saprykin, O.A. (2016) Control system of the anthropomorphous robot for work on the low-altitude earth orbit. *International Journal of Pharmacy and Technology*, 3, 18913-18199.
9. Hasegawa, T., Suehiro, T., Takase, K. (1992) A model-based manipulation system with skill-based execution. *IEEE Trans. Rob. and Autom.*, vol. 8, 535-544.
10. Pratt, J. , Dilworth P., Pratt, G. (1997) Virtual model control of a bipedal walking robot. *Proceedings of Int. Conf. on Robotics and Automation*, vol. 1, 193-198.
11. Hrr, J. Pratt, J.Chew, C.Herr, H.Pratt, G. (1998) Adaptive virtual model control of a bipedal walking robot. *IEEE Inter. Simps. Intellegence and Systems*, 245-251.
12. Tsukamoto, H., Takubo, T. ,Ohara, K., Mae, Y., Arai, T. (2010) Virtual impedance model for obstacle avoidance in a limb mechanism robot. (*ICIA*): *IEEE Inf. and Autom.*, 729-734.
13. Корендясев А. И., Саламандра Б. Л., Тывес Л. И. Манипуляционные системы роботов. М: Машиностроение, 1989.
14. Притыкин Ф. Н. Виртуальное моделирование движений роботов, имеющих различную структуру кинематических цепей. Омск: Изд-во ОмГТУ, 2014.
15. Whitney, D. E. (2010) The Mathematics of Coordinated Control of Prosthetic Arms and Manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 94 (4), 303-309.
16. Pritykin, F. N., Nebritov, V. I. (2021) Using weight values of generalized velocities to handle deadlocks in the synthesis of anthropomorphic robot arm movement. *Journal of Physics: Conference Series*, 1791(1), 012059.
17. Pritykin, F. N., Nebritov, V. I. (2019) Determination of target points approachability by an android robot arm in organized space based on virtual modeling of movements . *Journal of Physics: Conference Series*, 1260, 072015.

18. Притыкин Ф. Н., Небритов В. И. Графическая оптимизационная модель процесса сварки изделий роботом на чертеже Радищева // Программные системы и вычислительные методы. 2021. № 2. С. 63-73.
19. Притыкин Ф. Н. Моделирование гиперповерхностей, отражающих взаимосвязь кинематических параметров механизма андроидного робота // Вестник компьютерных и информационных технологий. 2022. Т. 19, № 7. С. 21-29.
20. Притыкин Ф. Н. Исследование кинематических параметров андроидного робота при автоматизированном синтезе движений по вектору скоростей // Омский научный вестник. 2022. № 2 (182). С. 5-9.

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Предмет исследования. Статья, исходя из названия, должна быть посвящена использованию компьютерного моделирования движения руки антропоморфного робота для определения положения его основания относительно объектов манипулирования. Содержание статьи соответствует заявленной теме.

Методология исследования базируется на анализе и синтезе данных, индукции и дедукции. Автор на высоком уровне использует графический инструментарий, что формирует крайне положительное впечатление от ознакомления с рецензируемой статьей. При доработке статьи и включении анализа данных о практике использования подобных роботов рекомендуется также построить график роста распространённости их применения в различных видах экономической деятельности. Это значительно расширит востребованность научной статьи у потенциальной читательской аудитории.

Актуальность исследования вопросов использования компьютерного моделирования движения руки антропоморфного робота для определения положения его основания относительно объектов манипулирования не вызывает сомнения, т.к. это напрямую отвечает поставленным Президентом России национальным целям развития, а также задачам по повышению технологического суверенитета.

Научная новизна в рецензируемой статье присутствует. В частности, она связана с представленным на рисунке 4 алгоритмом вычисления основания механизма руки антропоморфного робота. Представляется, что этот алгоритм будет интересен широкому кругу потенциальной читательской аудитории.

Стиль, структура, содержание. Стиль изложения научный. Структура статьи чётко обозначена, позволяет раскрыть содержание заявленных вопросов. Рекомендуется также добавить обсуждение полученных результатов исследования и расширить раздел «закключение», указав конкретные направления использования полученных результатов и дальнейшие векторы исследования. Так, автор утверждает, что полученные результаты «могут быть использованы при разработке информационно-управляющих комплексов подвижных объектов»: а кем конкретно? Какие эффекты от использования будут получены? В чём выгоды? Какие могут возникнуть потенциальные сложности и как на них реагировать? Ответы на эти вопросы значительно усилят данную научную статью и обеспечат её высокий уровень востребованности у широкого круга лиц, в т.ч. и за рубежом.

Библиография представлена 20 источниками. Ценно, что автор изучил и последние научные публикации. Также положительно характеризует статью по выбранной теме

исследования наличие зарубежных источников. Было бы интересно также дополнить анализом числовых данных практики использования подобных роботов в настоящее время и, соответственно, указать их источники.

Апелляция к оппонентам. Несмотря на сделанный краткий обзор источников во введении, какой – либо научной дискуссии не обнаружено. Было бы интересно изучить и сравнить взгляды разных учёных на поднимаемые вопросы, а также показать прирост научного знания по сравнению с тем, что уже содержится в научных публикациях.

Выводы, интерес читательской аудитории. С учётом всего вышеизложенного заключаем о высоком уровне написанной научной статьи. Устранение указанных в тексте рецензии замечаний позволит кратно повысить её востребованность у потенциальной читательской аудитории: и в научном сообществе, и в органах государственной власти, и в промышленности, и среди аналитиков.

Результаты процедуры повторного рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Предмет исследования являются, по мнению автора, обследование и анализ возможности и последствий использования компьютерного моделирования движения руки антропоморфного робота для определения положения его основания относительно объектов манипулирования.

Методология исследования исходя из анализа статьи можно сделать вывод о использовании в качестве методологической основы компьютерной диагностики моделировании движений роботов метод, основанный на использовании матриц частных передаточных отношений. Автором также использовался метод литературного анализа, сравнительных характеристик объектов и процессов алгоритмического и математического моделирования, метод построения положения целевых точек на горизонтальной проекции при различном положении объектов манипулирования в контейнере.

Актуальность затронутой темы связано с тем, что до настоящего времени исследований связанных с внедрением роботов в различных сферах деятельности человека в развитых странах мира недостаточно. Одно из направлений использования роботов является разработка так называемых «умных складов», основанных на использовании двух процессов - информатизации и роботизации. В сфере логистики ожидают к 2024 году достижения уровня использования робототехники для приема грузов, управления входящими запасами, упаковки исходящих грузов до 24%. При автоматизации работ в складских помещениях могут быть использованы антропоморфные роботы оснащенные системами управления движением с использованием элементов искусственного интеллекта, поэтому разработка интеллектуальных систем управления движением механизмами рук антропоморфных роботов способных заменить человека поведения робота в организованной среде является актуальной. Автор восполнил этот пробел.

Научная новизна заключается в попытке автора статьи на основе проведенных исследований компьютерного моделирования движения антропоморфного робота с использованием разработанного алгоритма показали возможность предвидеть и преодолевать тупиковые ситуации при компьютерном управлении движением с помощью изменения параметров взаимного расположения руки антропоморфного робота относительно объектов окружающей среды. Результаты могут быть использованы при разработке информационно-управляющих комплексов подвижных объектов, в частности при разработке систем управления автономно функционирующих антропоморфных

роботов в складских помещениях.

Стиль, структура, содержание стиль изложения результатов достаточно научный. Статья снабжена богатым иллюстративным материалом, отражающим процесс создания алгоритма компьютерного моделирования движения руки антропоморфного робота на фронтальной и профильной плоскостях проекций при установке объекта манипулирования в контейнер. Рисунки, фотографии и схемы иллюстративны, уместны.

Однако следует отметить ряд стилистических неточностей, повторов и грамматических ошибок, что впрочем не снижает научности представленных в статье результатов. (Например, Рис. 4 — Алгоритм вычисления вычисления основания механизма руки антропоморфного робота; использовании двух глобальных компонентов, это информатизация и робототизация; элементов искусственного интеллекта).

Библиография весьма исчерпывающая для постановки рассматриваемого вопроса.

Апелляция к оппонентам представлена в выявлении проблемы на уровне имеющейся информации, полученной автором в результате анализа.

Выводы, интерес читательской аудитории в выводах есть обобщения, позволяющие применить полученные результаты. Целевая группа потребителей информации в статье не указана.

Программные системы и вычислительные методы

Правильная ссылка на статью:

Макаров И.С., Ларин Д.В., Воробьева Е.Г., Емелин Д.П., Карташов Д.А. Влияние асинхронных и многопоточных моделей обработки запросов на производительность серверных веб-приложений // Программные системы и вычислительные методы. 2025. № 1. DOI: 10.7256/2454-0714.2025.1.73665 EDN: UZPPWT URL: https://nbpublish.com/library_read_article.php?id=73665

Влияние асинхронных и многопоточных моделей обработки запросов на производительность серверных веб-приложений

Макаров Игорь Сергеевич

ORCID: 0009-0004-8734-2667

кандидат технических наук

зав. кафедрой; кафедра прикладной информатики (ПИ); Поволжский государственный университет телекоммуникаций и информатики

443010, Россия, Самарская обл., г. Самара, Самарский р-н, ул. Льва Толстого, д. 23

✉ igor-psati@yandex.ru



Ларин Денис Вячеславович

ORCID: 0009-0000-1904-7210

студент; кафедра информатики и вычислительной техники (ИВТ); Поволжский государственный университет телекоммуникаций и информатики

443010, Россия, Самарская область, г. Самара, ул. Льва Толстого, 23

✉ denlar1989@gmail.com



Воробьева Евгения Григорьевна

ORCID: 0009-0008-8225-7091

студент; кафедра информатики и вычислительной техники (ИВТ); Поволжский государственный университет телекоммуникаций и информатики

443010, Россия, Самарская обл., г. Самара, Самарский р-н, ул. Льва Толстого, д. 23

✉ vorobeveva.g2004@gmail.com



Емелин Даниил Павлович

студент; кафедра информатики и вычислительной техники (ИВТ); Поволжский государственный университет телекоммуникаций и информатики

443010, Россия, Самарская область, г. Самара, ул. Льва Толстого, 23

✉ demelin163@gmail.com



Карташов Дмитрий Александрович

студент; кафедра информатики и вычислительной техники (ИВТ); Поволжский государственный университет телекоммуникаций и информатики

443010, Россия, Самарская обл., г. Самара, Самарский р-н, ул. Льва Толстого, д. 23

✉ tawerka40@gmail.com



[Статья из рубрики "Языки программирования"](#)

DOI:

10.7256/2454-0714.2025.1.73665

EDN:

UZPPWT

Дата направления статьи в редакцию:

12-03-2025

Аннотация: Объектом исследования являются серверные веб-приложения и их производительность при обработке большого количества одновременных запросов. В качестве предмета исследования рассматриваются асинхронные технологии (Node.js, Python Asyncio, Go, Kotlin Coroutines) и многопоточные модели (Java Threading, Python Threading). Авторы подробно анализируют асинхронные циклы событий, горутины, корутины и классические многопоточные подходы, оценивая их эффективность в задачах с интенсивным использованием I/O и вычислительных ресурсов. Проводится эксперимент с разработкой API на трёх языках (Java, Node.js, Go) и тестированием при помощи утилиты `hey`. Также исследуются особенности масштабируемости, оптимизации производительности, использование кэширования, обработка ошибок, нагрузочные тесты и особенности реализации параллельных вычислений. Цель исследования — определить, какие подходы обеспечивают наибольшую производительность в серверных приложениях. Методы исследования включают нагрузочное тестирование, сбор метрик (время отклика, пропускная способность и потребление ресурсов сервера) и анализ результатов. Научная новизна заключается в сравнении асинхронных и многопоточных методов в реальных сценариях веб-разработки. Основными выводами исследования являются рекомендации по использованию асинхронных технологий в высоконагруженных I/O задачах и многопоточности в вычислительно сложных сценариях. Полученные результаты помогут разработчикам оптимизировать производительность серверных приложений в зависимости от их задач и нагрузки. Дополнительно исследование рассматривает аспекты сложности отладки асинхронных приложений, влияние пулов потоков на производительность многопоточных решений, а также сценарии, в которых асинхронные и многопоточные подходы могут дополнять друг друга. Особое внимание уделено управлению ресурсами сервера при масштабируемых нагрузках, что позволит IT-специалистам более точно выбирать инструменты и технологии для решения конкретных задач. В заключении обсуждаются возможные пути оптимизации работы серверных приложений, включая использование новых подходов и алгоритмов, а также перспективы развития асинхронных и многопоточных технологий в контексте высоконагруженных систем, их влияние на общую архитектуру приложений, а также на повышение отказоустойчивости и безопасности.

Ключевые слова:

асинхронность, многопоточность, производительность, серверные веб-приложения, нагрузочное тестирование, Node.js, Python Asyncio, Go горутины, Java Threading, производительность серверных приложений

Введение

Современные серверные веб-приложения сталкиваются с растущими требованиями к обработке большого количества одновременных запросов. С увеличением числа пользователей и сложностью операций, особенно в приложениях, работающих с API, базами данных и внешними сервисами, производительность серверной части становится критически важной. В этом контексте выбор между асинхронными и многопоточными подходами приобретает большое значение.

Асинхронные технологии, такие как Node.js (цикл событий), Python Asyncio (параллелизм, управляемые события), Go (горутины), Kotlin Coroutines, предлагают возможности для улучшения производительности за счёт неблокирующих операций ввода-вывода и лёгкости управления большим числом одновременных соединений. С другой стороны, многопоточные модели, такие как Java Threading и Python Threading, обеспечивают надёжность и простоту логики, но могут иметь накладные расходы при создании большого числа потоков. [\[1, с.93-95\]](#), [\[2\]](#)

В данной статье рассматривается исследование производительности серверных веб-приложений с использованием асинхронных и многопоточных подходов.

2. Обзор технологий и подходов

Асинхронный цикл событий.

Асинхронный цикл событий представлен такими технологиями, как Node.js и Python Asyncio. Данный подход предполагает использование одного потока для управления большим количеством запросов через неблокирующий ввод-вывод. Это позволяет достичь высокой производительности при операциях ввода-вывода, однако усложняет отладку и требует использования асинхронных библиотек. [\[5\]](#)

Горутины и корутины.

Go и Kotlin Coroutines используют легковесные потоки, управляемые рантаймом языка. Они обеспечивают высокую параллельность и особенно эффективны в ресурсоемких задачах. Однако разработчику необходимо явно указывать конкурентные вычисления. [\[3, с.260-261\]](#), [\[4\]](#)

Многопоточное программирование.

Многопоточная модель (Java, Python Threading) выделяет отдельный поток для каждого запроса или использует пул потоков. Этот подход полезен для задач, требующих параллельной обработки, однако она требует внимательного подхода к проектированию и синхронизации, чтобы избежать ошибок и потерь производительности. [\[1, с.93-95\]](#)

Ниже приведена сравнительная таблица 2.1 данных технологий:

Таблица 2.1 – сравнение технологий и подходов

Подход	Технологии	Принцип работы	Преимущества	Недостатки
--------	------------	----------------	--------------	------------

Асинхронный цикл событий	Node.js, Python Asyncio	Один поток управляет большим количеством запросов через неблокирующий ввод-вывод	Высокая производительность при I/O операциях	Сложность отладки, необходимость адаптации асинхронным библиотекам
Горутины и корутины	Go, Kotlin Coroutines	Легковесные потоки, управляемые рантаймом языка	Высокая параллельность, эффективность в ресурсоемких задачах	Требуется понимания конкурентного программирования
Многопоточное программирование	Java Threading, Python Threading	Каждый запрос получает отдельный поток (или пул потоков)	Хорошая поддержка в языках, простота логики	Большие накладные расходы, создание потоков, блокировки ресурсов

3. Методология исследования

Для проведения исследования было разработано серверное приложение с API на трёх языках: Java (многопоточное), Node.js (асинхронное), Go (горутины). API-эндпоинт выполняет обработку входящего запроса, чтение данных из базы (эмуляция задержки) и отправку ответа. Нагрузочные тесты проводились с разным количеством одновременных запросов (50, 100, 500, 1000, 5000). Оценивались среднее время ответа, пропускная способность и использование ресурсов сервера (CPU, RAM).

4. Реализация

Нагрузочные тесты проводились для разных уровней нагрузки: 50, 100, 500, 1000 и 5000 одновременных запросов, что позволило выявить, как каждый подход справляется с увеличением нагрузки. Для каждого сценария тестирования мы измеряли следующие параметры:

Среднее время ответа — среднее время, которое прошло с момента получения запроса сервером до отправки ответа клиенту. Важный показатель для оценки отзывчивости приложения (таблица 4.1).

Таблица 4.1 – среднее время ответа

Язык	Низкая нагрузка (50 запросов)	Высокая нагрузка (5000 запросов)
Java	~304-306 мс (задержка + накладные расходы)	~1094-8966 мс (рост из-за блокировки потоков)
Node.js	~236-248 мс (почти без накладных расходов)	~401-785 мс (event loop справляется хорошо)
Go	~212-216 мс (немного выше из-за планировщика)	~384-549 мс (эффективное управление горутинами)

Node.js и Go имеют более низкую латентность под высокой нагрузкой благодаря асинхронной обработке и лёгкости создания новых задач.

Пропускная способность (RPS) — количество запросов, которые сервер способен обработать за единицу времени (таблица 4.2).

Таблица 4.2 – пропускная способность

Язык	Низкая нагрузка (50 запросов)	Высокая нагрузка (5000 запросов)
Java	~304-306 мс (задержка + накладные расходы)	~380-9087мс (рост из-за блокировки потоков)
Node.js	~233-249 мс (почти без накладных расходов)	~355-800 мс (event loop справляется хорошо)
Go	~211-217 мс (немного выше из-за планировщика)	~343-560 мс (эффективное управление горутинами)

Go лидирует из-за лёгкости горутин и быстрого планировщика, за ним идёт Node.js. Java замедляется из-за затрат на управление потоками и переключение контекста.

Для проведения нагрузочного тестирования нами была использована утилита *hey*.

hey — это современный инструмент для нагрузочного тестирования HTTP-серверов, написанный на языке Go. Его можно считать современным аналогом таких утилит, как *ApacheBench* или *wrk*, но с рядом преимуществ:

- Лёгкость и эффективность: *hey* позволяет быстро эмулировать тысячи одновременных запросов, что идеально подходит для проверки масштабируемости системы в условиях реальной нагрузки.
- Простота использования: для запуска теста достаточно указать общее число запросов и число одновременных соединений.

Интеграция утилиты *hey* в процесс нагрузочного тестирования позволила получить глубокое понимание поведения сервера под различными уровнями нагрузки, что является ключевым моментом при оптимизации веб-приложений и обеспечении их стабильной работы даже в условиях пиковых нагрузок. [\[6\]](#)

Использование CPU и RAM — сколько вычислительных ресурсов потребляет сервер при обработке запросов. Для их исследования была использована утилита *Process Explorer* (и также встроенного мониторинга PowerShell для конкретного процесса `Get-Process | Where-Object { $_.ProcessName -like "java" } | Select-Object Name, CPU, WS`). Этот инструмент показывает использование CPU, RAM и другую системную информацию для выбранного процесса (таблица 4.3). [\[7\]](#)

Process Explorer — это продвинутая утилита от Microsoft (Sysinternals), которая позволяет детально отслеживать активные процессы и анализировать их использование системных ресурсов, включая CPU (центральный процессор) и RAM (оперативная память). [\[7\]](#)

Таблица 4.3 – использование CPU и RAM

	RAM		CPU	
Язык	Низкая	Высокая	Низкая	Высокая

	нагрузка (50 запросов)	нагрузка (5000 запросов)	нагрузка (50 запросов)	нагрузка (5000 запросов)
Java	97,38 МБ	131,32 МБ	0,38%	6,82%
Node.js	65,79 МБ	110,73 МБ	0,38%	8,45%
Go	11,09 МБ	82,77 МБ	0,25%	3,80%

Go потребляет меньше всего памяти благодаря экономным горутинам. Node.js держится в пределах нормы, а Java требует больше памяти из-за тяжёлой модели потоков и работы JVM

Исходя из проведенных тестов, можно сделать вывод, что **Java** подходит для корпоративных приложений с комплексной логикой, но проигрывает по потреблению ресурсов и латентности при очень высокой нагрузке. **Node.js** отличен для I/O-зависимых задач и быстрой разработки. Лучшая производительность при средних нагрузках. **Go** идеален для высоконагруженных систем и требует меньше ресурсов. Выигрывает при большом количестве запросов.

Заключение

Исследование показало важность выбора подходящей модели обработки запросов для серверных веб-приложений в зависимости от специфики нагрузки. Асинхронные технологии, такие как Node.js и Go, продемонстрировали высокую эффективность при решении задач, связанных с интенсивным использованием ввода-вывода (I/O), обеспечивая стабильную производительность даже при высоких нагрузках. Эти технологии оказались более предпочтительными в сценариях с большим количеством одновременных запросов, где важно минимизировать задержки и ресурсоёмкость.

Многопоточные модели, такие как Java Threading и Python Threading, лучше подходят для вычислительно сложных задач, однако они сталкиваются с проблемами при высоких нагрузках из-за накладных расходов на создание и синхронизацию потоков. Это ограничивает их производительность в сценариях с большим количеством одновременных запросов.

Основные выводы исследования сводятся к следующим рекомендациям для разработки серверных приложений: для высоконагруженных I/O задач следует использовать асинхронные подходы, такие как Node.js и Go, в то время как для вычислительно интенсивных сценариев предпочтительнее использовать многопоточность. Также важно учитывать, что в некоторых случаях асинхронные и многопоточные модели могут дополнять друг друга, обеспечивая оптимальную производительность для различных типов нагрузки.

Особое внимание уделено управлению ресурсами сервера при масштабируемых нагрузках, что является ключевым аспектом для достижения высокой производительности и отказоустойчивости приложений. Использование правильных технологий и подходов в соответствии с типом задач позволит разработчикам более эффективно оптимизировать серверные приложения и повысить их безопасность.

Библиография

1. Опивалов С. А. Методы работы с потоками в языке Java // Международный журнал

гуманитарных и естественных наук. 2023. № 4 (79) Т. 3. С. 93-99.

2. Руководство по Node.js, часть 1: общие сведения и начало работы [Электронный ресурс]. URL: <https://habr.com/ru/companies/ruvds/articles/422893/> (Дата обращения: 03.03.2025).

3. Опивалов С. А. Перспективы использования языка Котлина в программировании // Международный журнал гуманитарных и естественных наук. 2023. № 9 (84) Т. 1. С. 260-262.

4. Параллельное программирование в Go [Электронный ресурс]. URL: <https://proglib.io/p/parallelnoe-programmirovaniye-v-go-2021-05-23?ysclid=m7wag43tb6712881695> (Дата обращения: 03.03.2025).

5. Asynchronous Functions and the Node.js Event Loop [Электронный ресурс]. URL: https://translated.turbopages.org/proxy_u/en-ru.ru.137d7d27-67c89218-0c37f6b9-74722d776562/https/www.geeksforgeeks.org/asynchronous-functions-and-the-node-js-event-loop/ (Дата обращения: 04.03.2025).

6. Load Testing using Hey [Электронный ресурс]. URL: <https://dev.to/saantoryuu/load-testing-using-hey-c84> (Дата обращения: 04.03.2025).

7. Process Explorer v17.06 [Электронный ресурс]. URL: <https://learn.microsoft.com/en-us/sysinternals/downloads/process-explorer> (Дата обращения: 02.03.2025).

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Тема особенно актуальна для облачных (cloud computing) и туманных (fog computing) глобально распределенных вычислительных систем, где нагрузка пользователей различных регионов мира характеризуется большими потоками данных параллельной обработки веб запросов (всемирная проблема big data), особенно на стороне сервера и баз данных.

Цель работы не сформулирована четко, хотя интуитивно понятно, что это повышение производительности веб приложений на стороне сервера в условиях повышенной нагрузки со стороны пользователей.

В работе приведен сравнительный обзор асинхронных и многопоточных подходов по критерию производительности серверных приложений. В качестве примеров асинхронных технологий рассмотрены Node.js, Python Asyncio, Go, Kotlin Coroutines, а многопоточных - Java Threading и Python Threading. Четко обозначены преимущества и недостатки каждой модели и приведены практические рекомендации их применения для разработки серверных приложений в зависимости от специфики нагрузки, что представляется как главная новизна работы.

Исследования доведены до программной реализации в виде нагрузочных тестов, что несомненно является достоинством данной работы. Приведены численные результаты среднего времени отклика и пропускной способности. Выявлено, что оба подхода справляются с увеличением пользовательской нагрузки. Незначительное отличие состоит в методах программирования, использования оперативной памяти и интеграции с другими программными системами.

Статья изложена грамотным техническим языком, понятным читательской аудитории в

данной предметной области. Структура оформления материала, введение, обзор технологий и подходов, методология исследований, реализация, заключение, библиография, соответствуют требованиям журнала. Таблицы приемлемого качества по критерию читабельности.

Критических замечаний не обнаружено. Работа может быть опубликована и соответствует тематике журнала "Программные системы и вычислительные методы".

Для улучшения качества работы и интереса читательской аудитории можно учесть следующие рекомендации:

1. Библиография содержит недостаточное количество источников 7, из них 5 это ссылки на Интернет ресурсы. Рекомендуется добавить именно научные публикации в журналах, желательно общедоступных.
2. В библиографии №1 грамматическая ошибка.
3. Ссылка в библиографии №5 не открывается.
4. Букву "ё" в тексте статьи не принято использовать, она может некорректно отображаться на сайтах Интернет и в системах цитирования, следует ее заменить на букву "е".
5. Фразы "мы измеряли" и "нами была использована" абстрактные, лучше конкретно указать кто, а лучше ссылку на источник информации.
6. Таблица 4.3 - надо уточнить единицу измерения, мегабит (Мб) или мегабайт (МБ).
7. Аббревиатура I/O расшифрована в заключении, а не при первом упоминании, и вообще необязательно расшифровывать I/O, CPU, RAM, т.к. они общеприняты в данной предметной области.

Общий вывод: статья может быть принята к публикации в журнале «Программные системы и вычислительные методы» с незначительными доработками.

Программные системы и вычислительные методы

Правильная ссылка на статью:

Золотухина Д.Ю. Сравнительный анализ индексационных стратегий в PostgreSQL при различных сценариях нагрузки // Программные системы и вычислительные методы. 2025. № 1. DOI: 10.7256/2454-0714.2025.1.73138
EDN: UUZPDW URL: https://nbpublish.com/library_read_article.php?id=73138

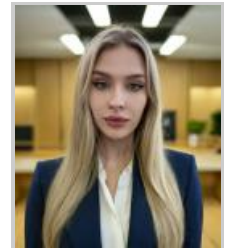
Сравнительный анализ индексационных стратегий в PostgreSQL при различных сценариях нагрузки

Золотухина Дарья Юрьевна

независимый исследователь

394062, Россия, Воронежская область, г. Воронеж, пер. Антокольского, 4

✉ dar.zolott@gmail.com



[Статья из рубрики "Показатели качества и повышение надежности программных систем"](#)

DOI:

10.7256/2454-0714.2025.1.73138

EDN:

UUZPDW

Дата направления статьи в редакцию:

24-01-2025

Аннотация: Предметом исследования является эффективность различных индексационных стратегий, реализованных в PostgreSQL, и их влияние на производительность операций SELECT, UPDATE и INSERT в условиях различных масштабов данных. Объектом исследования выступают индексы B-Tree, GIN и BRIN, применяемые для оптимизации работы баз данных. Автор подробно рассматривает такие аспекты темы, как временные характеристики выполнения операций, размер индексов и их ресурсоемкость. Особое внимание уделяется влиянию объема данных на производительность индексов и их пригодности для работы с различными типами данных, включая JSONB. Исследование направлено на систематизацию знаний о применении индексов для повышения эффективности работы высоконагруженных систем, где требуется оптимизация операций доступа, обновления и вставки данных, а также анализ потребления ресурсов. Ведущим методом исследования является эмпирический подход, включающий разработку тестовой базы данных с таблицами orders, customers и products. Эксперименты проводились для операций SELECT, UPDATE и INSERT на малых, средних и больших объемах данных. Для анализа использовались

метрики времени выполнения запросов и размера индексов, полученные с использованием инструментов PostgreSQL. Новизна исследования заключается в комплексном сравнении индексов B-Tree, GIN и BRIN в PostgreSQL с учетом не только временных характеристик выполнения запросов, но и их влияния на размер базы данных и общую нагрузку на систему. В отличие от существующих исследований, сосредоточенных на отдельных аспектах индексирования, данная работа рассматривает эффективность различных типов индексов в условиях изменяющейся нагрузки и различных категорий операций. Основными выводами проведенного исследования являются рекомендации по выбору индексов в зависимости от типов запросов и условий их выполнения. Индексы B-Tree подтвердили свою универсальность, демонстрируя высокую производительность для операций SELECT и UPDATE. GIN-индексы показали преимущества для работы с JSONB-данными, но их использование ограничено высокой ресурсоемкостью. BRIN-индексы доказали свою эффективность для больших объемов данных, особенно для операций SELECT. Особым вкладом автора в исследование темы является создание рекомендаций для разработчиков баз данных, что позволяет повышать производительность приложений за счёт оптимального выбора индексационной стратегии.

Ключевые слова:

индексационные стратегии, индексы B-Tree, индексы BRIN, индексы GIN, оптимизация запросов, производительность PostgreSQL, реляционные базы данных, JSONB-данные, высоконагруженные системы, управление данными

Введение

Современные реляционные базы данных, такие как PostgreSQL, предоставляют широкие возможности для оптимизации запросов благодаря разнообразным стратегиям индексирования. Эффективность индексов в значительной степени определяет производительность операций доступа, обновления и вставки данных, особенно в условиях высоких нагрузок и больших объемов информации [\[1\]](#). Несмотря на значительный прогресс в области индексирования, вопрос выбора оптимального типа индекса для конкретных сценариев работы остается актуальным. Это особенно важно для приложений, где тип операций, структура данных и масштаб нагрузки существенно варьируются.

Индексы B-Tree, GIN и BRIN, реализованные в PostgreSQL, представляют собой универсальные инструменты для решения различных задач. B-Tree индексы обеспечивают стабильную производительность и подходят для широкого спектра операций [\[2\]](#). GIN-индексы оптимизированы для сложных запросов с использованием JSONB-типов данных, что делает их востребованными в современных приложениях, работающих с неструктурированной информацией [\[3\]](#). BRIN-индексы, благодаря своей компактности, оказываются эффективными в сценариях с большими объемами данных и низкой интенсивностью обновлений [\[4\]](#). Однако различия в производительности этих индексов в зависимости от характера операций и масштабов данных требуют дополнительного исследования для выявления их оптимальных областей применения [\[5\]](#).

Объектом исследования являются индексационные стратегии, применяемые в PostgreSQL, с акцентом на три наиболее распространенных типа индексов: B-Tree, GIN и

BRIN. Эти индексы широко используются в высоконагруженных системах и оказывают значительное влияние на производительность баз данных.

Предметом исследования выступает эффективность применения различных индексов в зависимости от типа выполняемых операций (SELECT, UPDATE, INSERT), объема данных и особенностей их хранения. В работе проводится детальный анализ времени выполнения запросов, размера индексов и их влияния на производительность системы в разных сценариях нагрузки.

Настоящее исследование проводится с целью определения наилучших индексационных стратегий для PostgreSQL в условиях изменяющейся нагрузки. Работа направлена на систематизацию данных о времени выполнения операций и размерах индексов. Эксперименты осуществлялись на базе данных, содержащей таблицы orders, customers и products, которые моделируют реальные сценарии высоконагруженных систем. Полученные результаты представляют собой основу для рекомендаций по выбору индексационной стратегии, что особенно важно для повышения производительности приложений, работающих в условиях больших данных и высоких нагрузок.

Научная новизна исследования заключается в комплексном сравнении индексов B-Tree, GIN и BRIN в PostgreSQL с учетом не только временных характеристик выполнения запросов, но и их влияния на размер базы данных и общую нагрузку на систему. В отличие от существующих исследований [\[6, 7, 8\]](#), сосредоточенных на отдельных аспектах индексирования, данная работа рассматривает эффективность различных типов индексов в условиях изменяющейся нагрузки и различных категорий операций. Такой подход позволяет выявить закономерности их применения и сформулировать практические рекомендации по выбору оптимальной индексационной стратегии. Дополнительно проведен анализ зависимости размера индексов от увеличения объема данных, что даёт возможность прогнозировать рост инфраструктурных затрат при использовании различных стратегий индексирования.

Материалы и методы

В данном исследовании, направленном на анализ производительности различных типов индексов в PostgreSQL, эксперименты проводились на базе данных PostgreSQL версии 15.2. Эта версия была выбрана благодаря её поддержке широкого спектра индексов, включая B-Tree, GIN и BRIN, что позволило получить полное представление об их эффективности в различных сценариях использования [\[9\]](#).

Для построения тестовой среды была разработана база данных, включающая три таблицы: orders, customers и products. Структура таблиц и их наполнение были тщательно спроектированы с целью имитации реальных сценариев работы высоконагруженных систем. Таблица orders представляет собой список заказов, содержащий такие поля, как идентификатор заказа, идентификатор клиента, идентификатор товара, дата заказа и цена. Таблица customers хранит данные о клиентах, включая идентификаторы, имена и JSONB-метаданные о клиентах, что позволило протестировать индексы, специфичные для работы с JSONB-типами данных [\[10\]](#). Таблица products содержит информацию о продуктах, включая идентификатор, название и цену.

Таблицы заполнялись с использованием псевдослучайных данных для создания нагрузочных сценариев, максимально приближенных к условиям реальных систем. Например, даты заказов распределялись равномерно в пределах одного года, а цены продуктов варьировались в пределах от 1 до 1000 единиц. Для каждой таблицы были

созданы три варианта наполнения: малый объем (50 тысяч строк), средний объем (500 тысяч строк) и большой объем (5 миллионов строк). Это позволило оценить производительность индексов при разных масштабах данных. Данные были сгенерированы таким образом, чтобы исключить влияние аномалий и обеспечить репрезентативность результатов.

Эксперименты включали выполнение трех типов операций: SELECT, UPDATE и INSERT. Запросы были разработаны для оценки производительности индексов в различных сценариях. Пример запроса SELECT для таблицы orders:

```
SELECT *  
  
FROM orders  
  
WHERE order_date > '2023-01-01';
```

Для таблицы customers использовался JSONB-запрос:

```
SELECT *  
  
FROM customers  
  
WHERE interests @> '{"sports": true}';
```

Пример запроса UPDATE для таблицы products:

```
UPDATE products  
  
SET price = price * 1.1  
  
WHERE price BETWEEN 100 AND 500;
```

Пример запроса INSERT для таблицы orders:

```
INSERT INTO orders (customer_id, product_id, order_date, price)  
  
SELECT generate_series(1, 100000),  
  
random() * 1000,  
  
NOW() - interval '1 day' * random() * 365,  
  
random() * 1000;
```

В рамках исследования были протестированы индексы следующих типов:

1. B-Tree — универсальный индекс, применимый для большинства запросов [\[11\]](#).
2. GIN — предназначен для работы с JSONB-типами данных в таблице customers [\[12\]](#).
3. BRIN — эффективен для работы с большими объемами данных в таблицах orders и products [\[13\]](#).

Замеры производительности проводились для каждой комбинации индекса, объема данных и типа операции. Основные метрики включали:

1. Время выполнения — фиксировалось с использованием команды EXPLAIN ANALYZE и выражалось в миллисекундах (ms).
2. Размер индекса — измерялся с помощью функций PostgreSQL (pg_relation_size) и выражался в мегабайтах (MB).

Эксперименты проводились на машине со следующими характеристиками:

- Процессор: Apple M2 @ 3.49 GHz, 8 ядер, 8 потоков.
- Оперативная память: 8 ГБ.
- Накопитель: 256GB PCIe® NVMe™ M.2 SSD.

Для каждого эксперимента создавалась одинаковая тестовая среда, исключая влияние внешних факторов. Все замеры проводились многократно для получения точных и воспроизводимых результатов.

Полученные данные представлены в графической форме, что позволило проанализировать влияние объема данных, типа индекса и характера операции на производительность PostgreSQL.

Результаты

В ходе исследования были проанализированы временные показатели выполнения операций SELECT, UPDATE и INSERT на различных объемах данных (малый, средний, большой) с использованием индексов B-Tree, GIN, BRIN, а также в отсутствии индексов. Полученные данные представлены в графической форме, что позволяет наглядно оценить зависимость времени выполнения операций от объема данных и типа индекса.

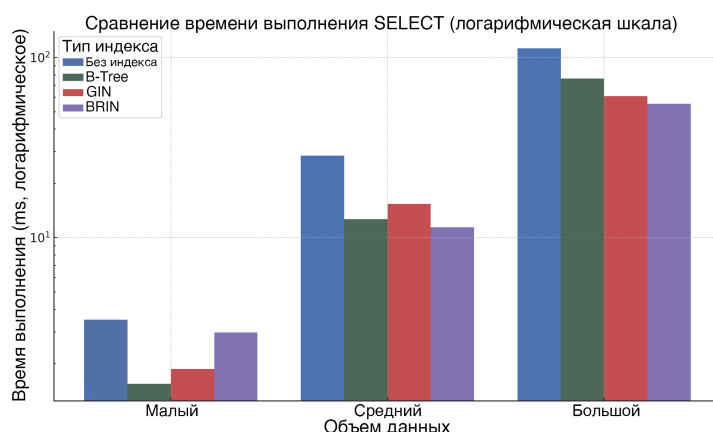


Рисунок 1. Влияние типа индекса и объема данных на время выполнения операции SELECT

На рисунке 1, отображающем результаты выполнения операции SELECT, видно, что при малом объеме данных использование индексов B-Tree, GIN и BRIN обеспечивает схожие временные характеристики, значительно превосходя выполнение без индекса. С увеличением объема данных эффективность GIN и B-Tree возрастает, однако индекс BRIN демонстрирует минимальное время выполнения для большого объема данных.

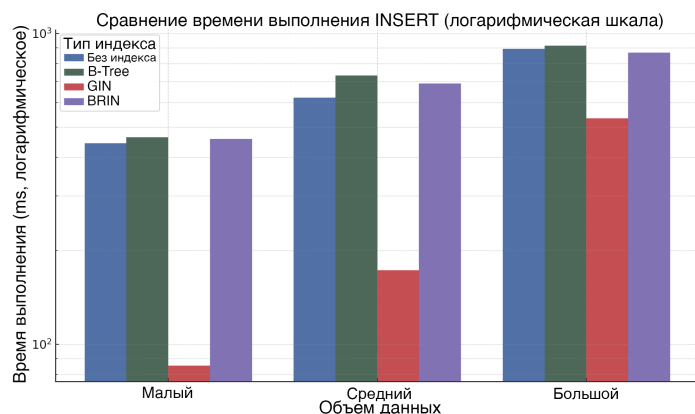


Рисунок 2. Влияние типа индекса и объема данных на время выполнения операции INSERT

Анализ операции UPDATE, который представлен на рисунке 3, показал, что при всех объемах данных наибольшую эффективность демонстрирует индекс B-Tree. GIN, напротив, увеличивает время выполнения операции на среднем и большом объемах данных. BRIN остается оптимальным выбором для большого объема данных, однако его преимущества становятся заметными только при значительных размерах таблиц.

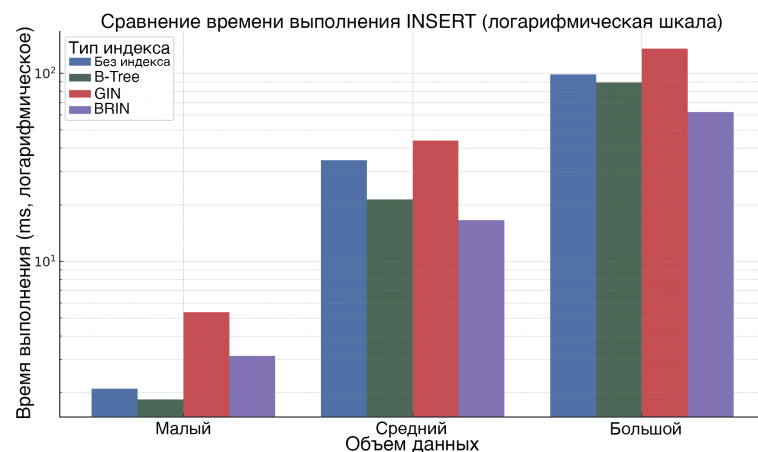


Рисунок 3. Влияние типа индекса и объема данных на время выполнения операции UPDATE

На рисунке 3 для операции INSERT видно, что индексы GIN и BRIN приводят к увеличению времени выполнения по сравнению с отсутствием индекса, особенно при большом объеме данных. Индекс B-Tree также несколько увеличивает время вставки, однако его влияние оказывается менее значительным.

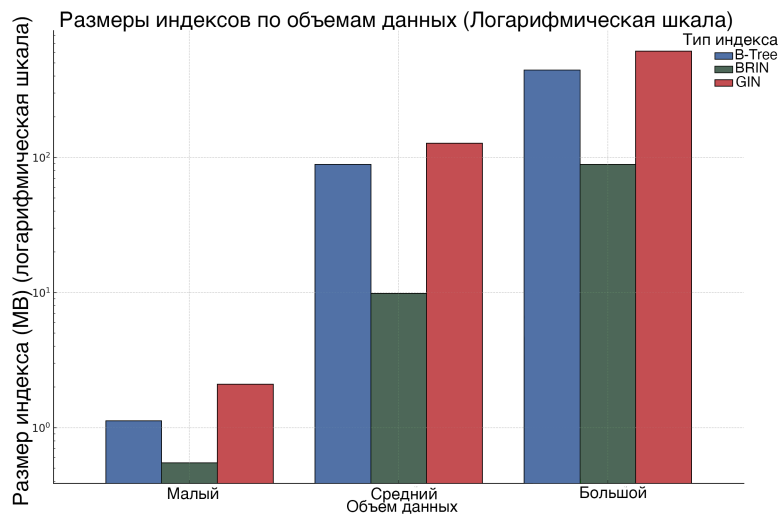


Рисунок 4. Размеры индексов различных типов в зависимости от объема данных

Размеры индексов зависят от их типа и объема данных, что отображено на рисунке 4. GIN-индексы демонстрируют наибольший размер, достигая 612.7 MB при большом объеме данных, тогда как B-Tree индексы занимают до 450.3 MB. BRIN-индексы остаются наиболее компактными, увеличиваясь до 92.1 MB при аналогичных условиях. Динамика изменения размеров индексов по мере роста объема данных показывает линейный рост для всех типов индексов, с наиболее заметным увеличением у GIN-индексов, что отражает их высокую ресурсоемкость при увеличении объема данных. B-Tree индексы растут более умеренно, сохраняя баланс между эффективностью и размером, а BRIN-индексы демонстрируют минимальные изменения, подчеркивая их экономичность в условиях больших объемов данных.

Обсуждение

Результаты экспериментов позволяют выявить характерные особенности каждого типа индекса в зависимости от объема данных и типа операции. Индексы B-Tree проявляют себя как универсальный выбор, демонстрируя стабильную производительность на всех объемах данных и для всех типов операций. Их эффективность при выполнении операций SELECT и UPDATE особенно заметна на больших объемах данных, где они сохраняют приемлемые временные характеристики без значительного увеличения времени выполнения.

Индексы GIN, несмотря на их превосходство в операциях SELECT с высокоиндексируемыми полями, оказываются менее подходящими для операций UPDATE и INSERT. Их ресурсоемкость и сложность обновления приводят к увеличению времени выполнения на средних и больших объемах данных. Эти особенности делают GIN-индексы релевантными в задачах, где частота обновления данных минимальна, а скорость выборки из сложных структур данных критична.

BRIN-индексы подтвердили свою эффективность для работы с большими объемами данных, особенно в операциях SELECT. Их компактная структура и минимальные накладные расходы на хранение делают их оптимальным выбором для анализа данных на уровне больших массивов. Однако их применение в операциях UPDATE и INSERT ограничено, так как их архитектура рассчитана на минимизацию издержек в сценариях, где обновление данных происходит реже, чем выборка.

Особое внимание заслуживают изменения в размерах индексов по мере роста объемов данных. Линейный рост, наблюдаемый у всех типов индексов, отражает их

предсказуемость и масштабируемость. GIN-индексы показывают наиболее значительное увеличение размера, что указывает на необходимость тщательного учета их использования в системах с ограниченными ресурсами. B-Tree индексы демонстрируют более умеренное увеличение размера, что делает их сбалансированным выбором для систем, где необходимо сочетать производительность и экономичность. BRIN-индексы остаются наиболее компактными, что подтверждает их пригодность для обработки больших объемов данных с минимальными затратами памяти.

Заключение

Результаты проведенного исследования демонстрируют значительное влияние индексационных стратегий на производительность PostgreSQL в условиях различных сценариев нагрузки и масштабов данных. Эксперименты показали, что выбор типа индекса является критически важным решением, напрямую влияющим на эффективность операций SELECT, UPDATE и INSERT, особенно при увеличении объема данных.

Индексы B-Tree подтвердили свою универсальность и стабильность производительности, что делает их оптимальным выбором для большинства сценариев. Эти индексы показали минимальное время выполнения операций SELECT и UPDATE на больших объемах данных, а также приемлемые характеристики для операций INSERT, что подтверждает их сбалансированность между эффективностью и ресурсопотреблением.

GIN-индексы проявили себя как эффективный инструмент для сложных выборок JSONB-данных, однако их ресурсоемкость и длительное время выполнения операций UPDATE и INSERT ограничивают их применение в сценариях с высокой частотой изменений данных. Эти особенности делают GIN-индексы подходящими для аналитических систем, где преобладают операции чтения сложных структурированных данных.

BRIN-индексы показали свои преимущества в работе с большими объемами данных, особенно для операций SELECT, благодаря компактности и минимальным накладным расходам на хранение. Однако их применение в операциях INSERT и UPDATE требует осторожного подхода, учитывая специфику их архитектуры, ориентированной на редкое обновление данных.

Анализ изменений размеров индексов по мере роста объема данных подтвердил их линейный характер, что свидетельствует о предсказуемости и масштабируемости всех типов индексов. Наиболее значительное увеличение размера было зафиксировано для GIN-индексов, что подчеркивает необходимость их взвешенного использования в системах с ограниченными ресурсами.

Результаты исследования предоставляют разработчикам и администраторам баз данных практические рекомендации по выбору индексационной стратегии в PostgreSQL. Эти рекомендации могут быть полезны для построения высокоэффективных систем, где требуется баланс между производительностью, ресурсопотреблением и спецификой приложения.

Перспективы дальнейшего исследования включают расширение анализа на другие типы индексов, такие как Hash-индексы и GiST, с целью оценки их применимости в различных сценариях нагрузки. Дополнительным направлением может стать исследование влияния многоверсионности (MVCC) в PostgreSQL на производительность индексов при высокой степени конкурентного доступа. Важной задачей также является разработка адаптивных стратегий индексирования, позволяющих динамически изменять структуру индексов в зависимости от нагрузки и типов запросов.

Библиография

1. Домбровская Г. Оптимизация запросов в PostgreSQL. М.: ДМК-Пресс, 2022.
2. Mostafa A. S. A Case Study on B-Tree Database Indexing Technique // Journal of Soft Computing and Data Mining. 2020. № 27-3. URL: <https://penerbit.uthm.edu.my/ojs/index.php/jscdm/article/view/6828> (дата обращения: 10.01.2025).
3. Borodin A., Mirvoda S., Porshnev S., Ponomareva O. Improving generalized inverted index lock wait times // Journal of Physics: Conf. 2018. №944. URL: <https://iopscience.iop.org/article/10.1088/1742-6596/944/1/012022/pdf> (дата обращения: 10.01.2025).
4. Borodin A., Mirvoda S., Kulikov I., Porshnev S. Optimization of Memory Operations in Generalized Search Trees of PostgreSQL // Communications in Computer and Information Science. 2017. № 716. URL: https://link.springer.com/chapter/10.1007/978-3-319-58274-0_19 (дата обращения: 10.01.2025).
5. Рогов Е. В. PostgreSQL 16 изнутри. М.: ДМК Пресс, 2024.
6. Морозов С. В., Нестеров С. А. Сравнительный анализ типов индексов в СУБД SQL Server и PostgreSQL // SAEC. 2024. № 2. С. 485–491.
7. Селиванов Е. О. Сравнение типов индексов в различных системах управления базами данных // Молодежь и наука: актуальные проблемы фундаментальных и прикладных исследований: Материалы VI Всероссийской национальной научной конференции молодых учёных. Комсомольск-на-Амуре: Комсомольский-на-Амуре государственный университет, 2023. С. 357–361.
8. Кудашов А. С., Агапова В. А., Дьячков Д. А., Казакова И. А. Обзор типов индексов и их применение в системах управления базами данных // Современные цифровые технологии. Материалы II Всероссийской научно-практической конференции. Барнаул: Алтайский государственный технический университет им. И. И. Ползунова, 2023. С. 299–303.
9. Documentation PostgreSQL 15 – URL: <https://www.postgresql.org/docs/15/release-15-2.html> (date of access: 10.01.2025).
10. Сорокин В. Е. Хранение и эффективная обработка нечетких данных в СУБД PostgreSQL // Программные продукты и системы. 2017. № 4. URL: <https://app.amanote.com/v4.1.10/research/note-taking?resourceId=Pprz23MBKQvf0Bhi37cu> (дата обращения: 10.01.2025).
11. B-tree индексы в базах данных на примере PostgreSQL. – URL: <https://techtrain.ru/talks/e2273ec8ca2b4ea692c65318a50c4be5> (дата обращения: 10.01.2025).
12. Меджидов Р. Г. Анализ многоколоночных индексов баз данных // Актуальные проблемы прикладной математики, информатики и механики : Сборник трудов Международной научной конференции. Воронеж : Научно-исследовательские публикации, 2019. С. 420–422.
13. Богатов И. В. Эффективная оптимизация запросов в СУБД Postgres / Академическая публицистика. 2022. № 5-2. С. 59–64.

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Представленная статья на тему «Сравнительный анализ индексационных стратегий в

PostgreSQL при различных сценариях нагрузки» соответствует тематике журнала «Программные системы и вычислительные методы» и посвящена актуальному вопросу выбора оптимального типа индекса для конкретных сценариев работы в условиях высоких нагрузок и больших объемов информации. Это обусловлено тем, что современные реляционные базы данных, такие как PostgreSQL, предоставляют широкие возможности для оптимизации запросов благодаря разнообразным стратегиям индексирования. Эффективность индексов в значительной степени определяет производительность операций доступа, обновления и вставки данных, особенно в условиях высоких нагрузок и больших объемов информации. Несмотря на значительный прогресс в области индексирования, вопрос выбора оптимального типа индекса для конкретных сценариев работы остается актуальным.

В статье представлен анализ литературных российских и зарубежных и интернет-источников по теме исследования. Указана теоретико-методологическая основа исследования. Авторами проведено исследование, направленное на анализ производительности различных типов индексов в PostgreSQL, эксперименты проводились на базе данных PostgreSQL версии 15.2. По мнению авторов данная версия выбрана с учетом поддержки широкого спектра индексов, включая B-Tree, GIN и BRIN. Эксперимент включал выполнение трех типов операций: SELECT, UPDATE и INSERT. Запросы были разработаны для оценки производительности индексов в различных сценариях.

Стиль и язык изложения материала является достаточно доступным для широкого круга читателей. Практическая значимость статьи четко обоснована. Статья по объему соответствует рекомендуемому объему от 12 000 знаков.

Статья достаточно структурирована - в наличии введение, заключение, внутреннее членение основной части (материалы и методы, результаты исследования и обсуждение).

Авторами проведено исследование, в ходе которого проанализированы временные показатели выполнения операций SELECT, UPDATE и INSERT на различных объемах данных (малый, средний, большой) с использованием индексов B-Tree, GIN, BRIN, а также в отсутствии индексов. Полученные данные представлены в наглядной графической форме. В заключении изложены основные результаты исследования, которые могут представлять интерес для разработчиков и администраторов баз данных с практической точки зрения с учетом изложенных авторами рекомендаций по выбору индексационной стратегии в PostgreSQL.

К недостаткам можно отнести следующие моменты: из содержания статьи не прослеживается научная новизна. Отсутствует четкое выделение предмета, объекта исследования.

Рекомендуется четко обозначить научную новизну исследования, сформулировать предмет, объект. Также будет целесообразным добавить о перспективах дальнейшего исследования.

Статья «Сравнительный анализ индексационных стратегий в PostgreSQL при различных сценариях нагрузки» требует доработки по указанным выше замечаниям. После внесения поправок рекомендуется к повторному рассмотрению редакцией рецензируемого научного журнала.

Результаты процедуры повторного рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Предмет исследования рецензируемой работы – эффективность применения различных индексов в российской версии бесплатной системы управления базами данных с открытым исходным кодом PostgreSQL при различных сценариях нагрузки в зависимости от типа выполняемых операций (SELECT, UPDATE, INSERT), объема данных и особенностей их хранения.

Методология исследования базируется на проведении экспериментов и сравнительном анализе времени выполнения запросов, размера индексов и их влияния на производительность системы в условиях изменяющейся нагрузки и различных категорий операций с применением графических методов представления полученных результатов.

Актуальность исследования авторы связывают с тем, что современные реляционные базы данных предоставляют широкие возможности для оптимизации запросов благодаря разнообразным стратегиям индексирования, которые влияют на производительность операций доступа, обновления и вставки данных, особенно в условиях высоких нагрузок и больших объемов информации, а также с недостаточной научной проработкой вопроса выбора оптимального типа индекса для конкретных сценариев работы.

Научная новизна рецензируемого исследования заключается в комплексном сравнении индексов B-Tree, GIN и BRIN в PostgreSQL с учетом не только временных характеристик выполнения запросов, но и их влияния на размер базы данных и общую нагрузку на систему.

В тексте статьи выделены следующие разделы: Введение, Материалы и методы, Результаты, Обсуждение, Заключение и Библиография.

В работе рассматривается некоммерческая версия объектно-реляционной системы управления базами данных, основанной на программе POSTGRES, разработанной на факультете компьютерных наук Калифорнийского университета в Беркли. В публикации акцент сделан на сравнительном анализе трех наиболее распространенных типов индексов, широко используемых в высоконагруженных системах и оказывающих значительное влияние на производительность баз данных: B-Tree, GIN и BRIN. В статье отражено влияние типа индекса и объема данных на время выполнения операций SELECT, INSERT, UPDATE; показаны размеры индексов различных типов в зависимости от объема данных. Продемонстрировано значительное влияние индексационных стратегий на производительность PostgreSQL в условиях различных сценариев нагрузки и масштабов данных. Эксперименты показали, что выбор типа индекса является критически важным решением, напрямую влияющим на эффективность вышеуказанных операций особенно при увеличении объема данных.

Библиографический список включает 13 источников – современные научные публикации отечественных авторов на русском и английском языках, а также интернет-ресурсы по рассматриваемой теме, на которые в тексте приведены адресные ссылки, что подтверждает наличие апелляции к оппонентам.

Рецензируемый материал соответствует направлению журнала «Программные системы и вычислительные методы», отражает результаты проведенной авторами работы, может вызвать интерес у читателей, рекомендуется к опубликованию.

Программные системы и вычислительные методы

Правильная ссылка на статью:

Макаров К.С., Фаткин Р.И. Скриншотное тестирование как многоаспектный вид автоматизированной динамической верификации веб-приложений // Программные системы и вычислительные методы. 2025. № 1. DOI: 10.7256/2454-0714.2025.1.73535 EDN: UVGEBC URL: https://nbpublish.com/library_read_article.php?id=73535

Скриншотное тестирование как многоаспектный вид автоматизированной динамической верификации веб-приложений

Макаров Константин Сергеевич

кандидат технических наук

зав. кафедрой; кафедра программного обеспечения и администрирования информационных систем;
Курский государственный университет

305000, Россия, Курская обл., г. Курск, Центральный округ, ул. Радищева, зд 33

✉ makarov_ks@kursksu.ru



Фаткин Руслан Игоревич

ORCID: 0009-0006-3481-6122

аспирант; кафедра программного обеспечения и администрирования информационных систем;
Курский государственный университет

305014, Россия, Курская обл., г. Курск, Центральный округ, ул. Рябиновая, д. 26Б, кв. 96

✉ ruslan4631@yandex.ru



[Статья из рубрики "Показатели качества и повышение надежности программных систем"](#)

DOI:

10.7256/2454-0714.2025.1.73535

EDN:

UVGEBC

Дата направления статьи в редакцию:

02-03-2025

Аннотация: Предметом исследования является многоаспектное скриншотное тестирование как современный метод автоматизированной динамической верификации веб-приложений, объединяющий функциональное тестирование и проверку пользовательского интерфейса (UI). Современные методы тестирования сталкиваются с проблемами высокой трудоемкости, ложноположительных срабатываний и низкой

масштабируемости, особенно в сложных проектах. Основная цель исследования — создать и внедрить метод, который позволяет повысить точность выявления дефектов, сократить время тестирования и снизить затраты на разработку тест-кейсов. В работе рассматриваются алгоритмы сравнения изображений, методы фильтрации динамических элементов и подходы к автоматизированному анализу интерфейсов для повышения эффективности и стандартизации в процессе верификации веб-приложений. В отличие от функционального и UI-тестирования по отдельности, предложенный метод позволяет анализировать несколько аспектов интерфейса и функциональности одновременно, что минимизирует трудозатраты и повышает надежность тестирования. Используется автоматическое сравнение эталонных и тестовых скриншотов на уровне пикселей, структурных элементов и содержимого с применением Python, Selenium, PIL и Pytest-xdist для параллельного выполнения тестов, что позволяет эффективно решать задачи верификации веб-приложений. Некоторые исследователи в области тестирования сходятся во мнении, что сам процесс тестирования мало стандартизирован и не имеет четких критериев оценки его эффективности. Предлагаемый нами метод позволяет добиваться решения поставленных задач верификации даже в условиях изменяющихся стратегий и подходов к оценке работоспособности системы за счет создания гибкой и точной системы проверки, которая объединяет различные типы тестирования в единую структуру, что делает данный метод подходящим для современных задач разработки программного обеспечения. Экспериментальная часть демонстрирует преимущества многоаспектного скриншотного тестирования по сравнению с другими методами, включая сокращение времени тестирования, повышение точности обнаружения дефектов и улучшение анализа получаемых отчетов. Данный подход может быть адаптирован к различным сценариям тестирования и выгоден для использования в высоконагруженных проектах, требующих регулярной регрессионной проверки.

Ключевые слова:

скриншотное тестирование, многоаспектное скриншотное тестирование, автоматизированное тестирование, динамическая верификация Веб-приложения, Python в тестировании, Selenium и PIL, UI тестирование, оптимизация тестирования, критерии тестирования, задачи верификации

Введение

В процессе разработки любого программного обеспечения (ПО) важным этапом является тестирование создаваемых решений. Современные приложения, особенно веб и мобильные, требуют регулярного обновления и доработки в условиях быстро меняющихся требований как со стороны рынка, так и со стороны внедрения новых технологий. Для обеспечения высокого уровня качества и поддержания стабильного функционирования важно регулярно проводить анализ состояния системы т. к. даже малозначительные ошибки могут привести к потере стабильности, ухудшению пользовательского опыта и т. д., а как следствие к экономическим издержкам. Так, ежегодные экономические потери из-за некачественного ПО только в США оцениваются в десятки миллиардов долларов [\[1, с. 14\]](#), что подчеркивает актуальность разработки надежных методов тестирования программного обеспечения.

Веб-приложения стали неотъемлемой частью жизни для миллиардов людей по всему миру. И большинство таких решений имеют коммерческую направленность. Для продукта, целью которого в конечном счете является получение прибыли, экономические потери,

связанные с некачественным ПО, могут стать одним из главных источников проблем т.к. любые ошибки, приводящие к ухудшению опыта использования, ведут к потерям в количестве реальных пользователей. Именно тестирование выпускаемого продукта призвано решить ряд таких проблем.

Скриншотное тестирование, ориентированное на автоматическое сравнение интерфейсов, активно применяется в динамической верификации ПО. Однако традиционные подходы, такие как функциональное или отдельное UI-тестирование, имеют ограничения: высокие трудозатраты, избыточное количество ложных срабатываний и низкую производительность при масштабировании задач. В случае с высоконагруженными и сложными веб-приложениями отслеживать все эти процессы в ручном режиме достаточно трудоемкая задача, которая чревата ошибками со стороны тестировщика.

Гурин Р. Е., Рудаков И. В., Ребриков А. В. рассматривают существующие подходы к верификации ПО и приводят анализ их ограничений и эффективности^[2]. Проблемы, выделенные в статье, указывают на необходимость новых решений, способных преодолеть ограничения традиционных подходов.

Также, в исследованиях^[3-5] отмечается, что верификация программного обеспечения имеет ряд ограничений, связанных с:

1. ограничением времени на тестирование и выделенным бюджетом;
2. человеческим фактором (субъект, проводящий тестирование, напрямую влияет на его результат);
3. изменение требований к работе ПО уже в процессе разработки или на этапе поддержки продукта;
4. формулировкой логики задач верификации напрямую влияет на получаемые результаты, если задача сформулирована неполно или некорректно, результат можно признать недействительным;
5. отсутствие современных метрик для оценки эффективности тестирования.

Проблемы, выделенные в исследованиях, указывают на необходимость новых решений, способных преодолеть ограничения традиционных подходов. Многоаспектное скриншотное тестирование предлагает комбинированный подход, объединяющий функциональное и UI-тестирование, что может повысить точность и производительность верификации ПО.

Кудрявцева Е. Ю., также отмечает, что использование автоматизированного тестирования в больших проектах наиболее обосновано и релевантно^[6]. Таким образом, предлагаемая методика отвечает на вызовы и способствует дальнейшему развитию области верификации программного обеспечения.

Стоит также отметить, что в отечественной научной литературе на данный момент вопросы, связанные с верификацией ПО, в частности, общедоступного назначения, рассматриваются редко или ограничиваются обзорными статьями существующих методов. Ввиду этого область тестирования ПО развивается не столь стремительно, в отличие от разработки. Однако быстрые темпы развития области разработки ПО требуют от методов верификации такой же быстрой адаптации для решения актуальных задач. Так, целью данной работы было определено следующее — разработать метод многоаспектного

скриншотного тестирования, сочетающего функциональные и UI-подходы, для улучшения точности выявления дефектов и оптимизации ресурсов в процессе верификации веб-приложений. Для достижения цели предложено использовать язык программирования Python с библиотеками Selenium и PIL, что обеспечивает автоматизацию тестов и снижение трудозатрат.

1. Методы исследования

Методология исследования основана на разработке и применении многоаспектного скриншотного тестирования, которое представляет собой комбинацию функционального и UI-тестирования, позволяющую эффективно решать задачи верификации веб-приложений. Этот метод позволяет одновременно проверять несколько аспектов работы программного обеспечения в одном запуске тестового сценария. Такой подход призван снизить количество ложных срабатываний и увеличить точность тестирования.

Основная идея метода заключается в проведении параллельного анализа функциональных и визуальных характеристик веб-приложения путем автоматического сравнения эталонных и тестовых скриншотов. Это достигается за счет:

1. Создания эталонных версий скриншотов, которые фиксируют корректное состояние визуальных и функциональных элементов.
 2. Разработки тестовых сценариев, направленных на автоматизированное сравнение текущего состояния приложения с эталонным по нескольким уровням:
- **Пиксельный уровень:** проверка различий между эталонным и тестовым скриншотами на уровне отдельных пикселей.
 - **Уровень структурных элементов:** анализ корректности отображения интерфейсных элементов (размеры, расположение, отступы, шрифты).
 - **Уровень содержимого:** проверка соответствия текстовой и графической информации эталону.

Методология ориентирована на минимизацию ложноположительных результатов, которые могут возникать при динамических изменениях интерфейса (например, всплывающих уведомлений или анимациях). Для этого внедрены фильтры и маскирование динамических элементов, что позволяет исключить их влияние на итоговые результаты тестирования.

Экспериментальная часть методологии направлена на проведение сравнительного анализа многоаспектного скриншотного тестирования с традиционными методами, такими как функциональное и UI тестирование. Для этого были выделены ключевые метрики оценки:

- Количество выявленных дефектов;
- Точность тестирования (учитывая ложные срабатывания и пропущенные ошибки);
- Время выполнения тестов;
- Трудозатраты на разработку и поддержку тестов;
- Надежность и стабильность тестов.

Методология была протестирована на веб-приложении, разработанном в экспериментальных целях, и охватывала такие аспекты, как:

- Проверка отображения форм (регистрация, авторизация);
- Проверка работы элементов интерфейса (кнопки, поля ввода);

- Адаптивное отображение интерфейса в различных конфигурациях (тёмная/светлая темы, разрешение экрана).

Используемая методология обеспечивает более глубокий анализ взаимодействия элементов интерфейса, что позволяет выявлять дефекты, которые могли быть пропущены при применении других видов динамической верификации.

Так, многоаспектное скриншотное тестирование направлено на конфигурируемую проверку нескольких тестовых ситуаций для заданной верификационной задачи за один запуск автоматизированных тестовых сценариев. Каждый аспект сосредоточен на верификации определенных функциональных или визуальных элементов интерфейса и, в процессе прохождения теста, должны быть обнаружены все нарушения для проверяемых тестовых ситуаций. Тестовыми ситуациями называются те ситуации, в которых выполняется тестирование, а процедуры, описывающие процесс создания этих ситуаций и проверки, которые необходимо выполнить над полученными результатами, — тестами [\[7, С. 68\]](#). В рамках верификации ПО путем многоаспектного скриншотного тестирования рассматриваются функциональные и UI характеристики объекта.

Объектом исследования в данной работе является веб-приложение, которое требует регулярной проверки для обеспечения стабильной работы и соответствия требованиям пользователей. Таким образом, был выбран интернет-магазин, разработанный на платформе WordPress, с акцентом на страницу «Мой аккаунт». Приложение является типичным представителем коммерческих веб-ресурсов, которые широко используются для предоставления онлайн-услуг. Оно содержит стандартные элементы пользовательского интерфейса, такие как формы для ввода данных, кнопки, чек-боксы, выпадающие списки и визуальные элементы (иконки, цветовые схемы). Страница авторизации и регистрации является критически важной для любого интернет-магазина, так как обеспечивает доступ к функционалу приложения и влияет на пользовательский опыт. В качестве тестовой среды использовалась имитация различных реальных условий эксплуатации: разные разрешения экрана, светлые и темные темы интерфейса, а также различные браузеры и устройства (десктопные и мобильные). Применение многоаспектного скриншотного тестирования для такого объекта позволяет эффективно продемонстрировать преимущества метода, включая выявление сложных ошибок, которые могли бы остаться незамеченными при использовании только функционального или UI-тестирования.

Таким образом, выбранный объект исследования — это универсальный пример, который позволяет оценить эффективность и применимость предложенного метода для более широкого спектра веб-приложений.

Для реализации предложенной методологии многоаспектного скриншотного тестирования использовался комплекс инструментов и программных библиотек, обеспечивающих автоматизацию процессов верификации, обработки изображений и анализа результатов. Для упрощения реализации многоаспектного подхода предложенный метод был интегрирован с языком программирования высокого уровня Python. Благодаря своей гибкости и обширной библиотеке инструментов для работы с изображениями и пользовательскими интерфейсами, язык предоставляет все необходимые средства для реализации многоаспектного подхода [\[8\]](#). Таким образом, были использованы:

1. библиотеки Python, такие как Selenium для взаимодействия с веб-интерфейсами;

2. PIL для захвата и обработки скриншотов, включая сравнение эталонных изображений с текущими версиями;
3. Pytest-xdist для обеспечения параллельной обработки тестовых сценариев, позволяющие запускать тесты на нескольких конфигурациях устройств или браузеров одновременно.
4. JavaScript для маскирования динамических элементов интерфейса (например, анимаций или всплывающих окон), чтобы избежать ложных срабатываний при сравнении изображений.

Перечисленные технологии хорошо интегрируются в единую систему, поддерживают настройку под различные устройства, браузеры и темы интерфейса, а также позволяют параллельно выполнять тесты, что сокращает время на верификацию. Это позволило в рамках работы автоматизировать процесс скриншотного тестирования, который одновременно проверяет несколько аспектов ПО.

Также стоит отметить, что многоаспектное скриншотное тестирование повышает шансы на выявление потенциальных дефектов, которые могли бы остаться незамеченными при использовании других видов динамической верификации, например, функционального тестирования, где проверку проходит возможность взаимодействия с системой. Различные аспекты могут вести себя по-разному в зависимости от связи с другими элементами системы или условиями отображения. Например, при взаимодействии или наведении на кнопку ее цвет должен меняться, но функциональное тестирование призвано проверить только то, что с кнопкой можно взаимодействовать. Именно одновременная проверка позволяет выявлять и анализировать сложные взаимосвязи и взаимодействия между разными аспектами ПО, ошибки в которых могли бы остаться незамеченными при поэтапном тестировании отдельных аспектов. Например, при многоаспектном скриншотном тестировании веб-приложений можно одновременно проверять расположение кнопок, заполняемость инпутов, цветовые схемы и состояние различных элементов интерфейса, что делает тестирование более полным и менее подверженным ошибкам из-за невнимательности или кроссплатформенности.

2. Описание предлагаемого метода

Задача верификации была сформулирована так, что все функциональные и нефункциональные тесты были объединены в один поток, с помощью которого проведена проверка наибольшего числа аспектов ПО за наименьшее время прохождения тест-кейсов. Так, многоаспектное скриншотное тестирование даст возможность иметь сотни декомпозированных тест-кейсов в одном сценарии, что позволяет добиться хороших результатов производительности и экономической выгоды.

Каждый тест-кейс имеет свою задачу верификации, которая определяет корректное поведение. Так, путем сравнения скриншотов эталонной версии ПО с текущей версией, система анализирует интерфейс на предмет совпадения/несовпадения. В случае нахождения несовпадения тест-кейс отдает false и показывает метку ошибки в точке расхождения скриншота с ожидаемым поведением ПО. Все это фиксируется с помощью алгоритма вывода трасс-ошибок.

Трасс-ошибки (или trace-ошибки) — это последовательность шагов или операций, выполненных системой или программой, которые привели к возникновению ошибки. Они помогают локализовать сбои, что упрощает процесс диагностики и исправления ошибки.

Результат задачи верификации может иметь одно из следующих состояний:

- **Успешно (Passed):** функциональные и UI аспекты соответствуют ожидаемым результатам на всех скриншотах, что подтверждает корректность работы в рамках поставленной задачи.
- **Ошибка (Failed):** произошло несоответствие, и была зафиксирована ошибка. Трасс-ошибки позволяют определить, какие шаги привели к сбою, помогая быстрее устранить проблему.

Основные этапы процесса разработки и внедрения многоаспектного скриншотного тестирования можно представить в виде списка:

- **Подготовка тестовой среды**

На этом этапе необходимо определить, какие аспекты ПО будут тестироваться. В данном случае: переходы; заполнение форм; регистрация и авторизация; разрешение экрана; ориентация; темы (тёмная/светлая); браузеры или платформы (мобильные и десктоп устройства).

Также важно сформулировать задачу динамической верификации перед созданием тестовых сценариев. Например, проверка правильности отображения визуальных элементов, текста, иконок и т.д и возможности взаимодействия с ними. После этого стоит развернуть тестовую среду, которая будет максимально приближена к реальным условиям использования. Это включает настройку симуляторов устройств или браузеров, использование различных конфигураций экранов, загрузку необходимых шрифтов и библиотек.

Для каждого из аспектов, задействованного в скриншотном тестировании, можно использовать разные уровни проверки. Например, можно выполнить более глубокую проверку только тех элементов, которые уже были идентифицированы как потенциально проблемные на начальном этапе анализа. Это позволяет уменьшить избыточные затраты ресурсов.

- **Создание эталонных скриншотов**

Под эталонными скриншотами мы понимаем те изображения, которые отражают желаемое или правильное состояние интерфейса. Именно они будут служить основой для последующего сравнения. Скриншоты делаются на основе той версии программы, которую мы принимаем за эталонную (она либо ранее была протестирована и отлажена, либо используется дизайн-макет системы). При создании скриншотов предлагается опираться на те технические требования к средам и устройствам, которые были определены на предыдущем шаге. В рамках задачи верификации при подготовке тест-кейсов можно выделить несколько конкретных аспектов ПО, которые будут требовать проверки по различным параметрам. В рамках многоаспектного скриншотного тестирования параметры можно разделить следующим образом:

- Глобальные отклонения – это значительные изменения в ожидаемом поведении ПО (например, исчезновение элементов, серверные ошибки).
- Детальные отклонения – небольшие изменения, такие как изменение цвета или незначительное смещение элементов.
- Функциональные отклонения – ошибки в работе элементов, с которыми можно взаимодействовать.

- **Автоматизированное снятие тестовых скриншотов**

После сбора технических требований и подготовки эталонных версий скриншотов, запускаются автоматизированные тестовые сценарии, которые делают снимки текущего состояния ПО в тех же условиях, в которых были сделаны эталонные. Для многоаспектного скриншотного тестирования важно предусмотреть создание определенных условий для проверки. Например, проверку разных тем оформления, языков локализации, устройств с различными экранами и разрешениями. Это позволяет расширить покрытие тестирования и убедиться в корректной работе ПО в различных ситуациях.

• Многоаспектное сравнение

После автоматизированного снятия тестовых скриншотов на текущей версии ПО, автоматизированные тестовые сценарии начинают сравнивать два имеющихся в библиотеке скриншота на соответствия/несоответствия. Сравнение проходит на трех уровнях:

- пиксельный уровень, на котором определяются любые различия между эталонными и тестовыми скриншотами (изменение позиций, шрифтов, цветов).
- уровень проверки по структурным элементам позволяет проверить корректность отображения элементов интерфейса. Это могут быть размеры маржинов/падингов, кнопок и инпутов, их расположение во вьюпорте.
- уровень проверки по содержимому, при котором проверяется не только соответствие размеров/цветов шрифта, но и корректность самого содержания. Помимо текстовой информации, можно также проверить, что все графические элементы интерфейса отображаются корректно и находятся на своих местах.

• Отчетность и анализ результатов

После завершения тестов необходимо задать параметр запуска с использованием фреймворка для формирования отчетности с описанием всех негативных и позитивных исходов. Эти отчеты могут включать скриншоты с подсвеченными проблемными участками, что упрощает анализ и понимание проблемы со стороны как тестировщика, так и разработчика.

На основе полученных отчетов команды могут принимать решения о необходимости исправления тех или иных проблем. Это может быть анализ вручную для сложных случаев или автоматическое принятие решения для очевидных отклонений.

Для сокращения ложноположительных срабатываний может потребоваться настройка условий фильтрации полученных результатов. Так, все найденные отклонения должны быть классифицированы по их критичности. Например, исчезнувшие элементы верстки или неработающая форма регистрации могут быть намного важнее, чем незначительные отклонения в оттенке кнопки или ее сдвиг на несколько пикселей. Также стоит обратить внимание на динамически изменяющиеся элементы: всплывающие окна, анимации, номера телефонов с автозаменой и т.д. В предлагаемом методе для визуальной проверки интерфейса используется библиотека PIL для обработки изображений, а Javascript помогает маскировать динамические элементы, такие как всплывающие уведомления или анимации, которые могут меняться между запусками и мешать корректному сравнению. Это обеспечивает более детализированное и точное сравнение, исключая нежелательные ошибки, которые не относятся к функциональным изменениям.

Таким образом, для минимизации ложноположительных срабатываний нужно настраивать

уровни чувствительности тестов или проработать алгоритм обхода анализа нежелательных к сравнению элементов. Это поможет сосредоточиться на критических багах и избежать отражения в отчетах незначительных проблем, которые не влияют на функционирование системы.

- **Рефакторинг и оптимизация тестов**

Если приложение изменилось в соответствии с новыми требованиями, старые эталонные скриншоты могут стать неактуальными. В таком случае их нужно обновить, чтобы будущие тесты были релевантны текущему состоянию ПО. Это важный процесс, который помогает избежать ложных срабатываний или несоответствия поставленных задач верификации. Также может потребоваться оптимизация тестовой системы. Постоянное улучшение алгоритмов сравнения скриншотов и инструментов автоматизации — это ключевой момент для повышения эффективности тестирования. Например, можно улучшить систему фильтрации результатов или внедрить более точные методы анализа. Оптимизация позволяет ускорить тестирование и сократить время на ручную проверку результатов.

Также в некоторых случаях дополнительно можно реализовать итерационный процесс верификации, при котором проверка правильности или соответствия системы, программы, либо ее отдельных компонентов выполняется многократно после каждого этапа разработки. Например, одна из команд разработки занимается новой функцией, которую планируют интегрировать в уже созданный продукт, в таком случае нет нужды ждать, пока новая функция появится в конечном продукте — можно начинать тестировать ее сразу. Это позволит выявлять ошибки или несоответствия требованиям на ранних стадиях и вносить необходимые изменения перед переходом к следующей фазе работы или интеграцией с основным проектом. В таком случае, при последующем запуске многоаспектного скриншотного тестирования сократится количество ошибок, которые можно было выявить раньше, что позволит сосредоточиться на проверке взаимодействия новых и старых элементов в системе.

3. Экспериментальная апробация многоаспектного скриншотного тестирования

В рамках работы был проведен эксперимент, целью которого была проверка следующей гипотезы: многоаспектное скриншотное тестирование является более эффективным видом верификации веб-приложений по сравнению с функциональным и UI тестированием, благодаря гибриднему анализу всех аспектов ПО в рамках выбранного тестового сценария за один запуск тестов.

Цель эксперимента: доказать, что многоаспектное скриншотное тестирование является более эффективным видом верификации.

В рамках эксперимента был создан сайт интернет-магазина на WordPress для облегчения документирования и анализа полученных результатов.

Задача верификации была поставлена следующим образом: необходимо проверить работоспособность веб-страницы интернет-магазина «Мой аккаунт», которая предназначена для прохождения авторизации и регистрации пользователей. Под работоспособностью мы понимаем следующее: корректность отображения страницы согласно эталонным версиям ПО, возможность функционального взаимодействия с различными элементами, корректная передача информации (логин/пароль).

Оценка эффективности проводилась методом сравнительного анализа на примерах функционального, ui и многоаспектного скриншотного тестирования.

Критерии оценки эффективности: эффективность созданных решений оценивалась по следующим критериям:

- Количество выявленных дефектов: оценка эффективности тестирования на основе количества обнаруженных отклонений, включая графические артефакты и функциональные ошибки. **Метрика:** Общее количество обнаруженных дефектов каждым из видов тестирования. Сравнение между видами в процентном соотношении.
- Время выполнения тестов: анализ времени, затраченного на проведение тестов каждым из методов. **Метрика:** Среднее время, затраченное на выполнение каждого тестового сценария в рамках метода.
- Трудозатраты на разработку и поддержку тестов: оценка сложности разработки тестов и их поддержки на каждом этапе. **Метрика:** Количество часов, затраченных на разработку и поддержку тестов в разрезе каждого метода. Измерялось в процентном соотношении.
- Точность тестирования: сравнение точности обнаружения ошибок при каждом методе тестирования, включая ложные срабатывания и пропущенные дефекты. **Метрика:** Процент ложных срабатываний и пропущенных дефектов
- Надежность: оценка стабильности тестов, измеряемая количеством непредсказуемых отказов тестов.

В рамках верификационной задачи были определены следующие тестовые сценарии:

- для функционального тестирования — 2 тестовых ситуации (Табл. 1);
- для UI тестирования — 16 тестовых ситуаций (Табл. 2);
- для многоаспектного скриншотного тестирования — 4 тестовых ситуации (Табл. 3).

Рассмотрим их подробнее:

Таблица 1
Тестовая модель функционального тестирования

Номер кейса	Название теста	Шаги	Ожидаемый результат
1	Проверка формы регистрации. Регистрация пользователя	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается форма авторизации
		2. Выбрать в меню "Зарегистрироваться"	Отображается форма регистрации
		3. Заполнить Поля: Поле почты ФИО Пароль Подтверждение пароля Чекбоксы Кнопка регистрации	Поля успешно заполнены
		4. Нажать на кнопку Зарегистрироваться	Регистрация выполнена успешно
2	Проверка формы авторизации. Логин	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается

	пользователя		форма авторизации
		2. Заполнить поля Имя пользователя	Имя успешно введено
		3. Заполнить поле Пароль	Пароль введён
		4. Клик на кнопку Войти	Вход успешно произведён

Таблица 2

Тестовая модель UI тестирования

Номер кейса	Название теста	Шаги	Ожидаемый результат
1	Проверка формы регистрации (UI)	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается форма регистрации
		2. Выбрать в меню "Зарегистрироваться"	Отображается форма регистрации
		3. Проверка полей в форме регистрации	Отображаются: Поле почты ФИО Пароль Подтверждение пароля Чекбоксы Кнопка регистрации
2	Проверка формы авторизации UI	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается форма авторизации
		2. Проверка отображений меню для входа	Отображается Почта или логин Пароль Ссылка Забыли пароль Чекбокс Запомнить меня Вкладка Вход
3-10	Проверка формы регистрации (UI). Проверка свойств	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается

полей: Поле почты		форма регистрации
Фамилия	Проверка свойств css поля почты	Свойства выбранного элемента
Имя		отображаются
Пароль		корректно
Подтверждение пароля		относительно заданных
Чекбокс 1 и Чекбокс 2		
Кнопка регистрации		

Таблица 3

Тестовая модель многоаспектного скриншотного тестирования

Номер кейса	Название теста	Шаги	Ожидаемый результат
1	Проверка формы регистрации (UI)	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается форма авторизации
		2. Выбрать в меню "Зарегистрироваться"	Отображается форма регистрации
		3. Проверка полей в форме регистрации	Отображаются: Поле почты ФИО Пароль Подтверждение пароля Чекбоксы Кнопка регистрации
2.	Проверка формы регистрации (UI) - регистрация	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается форма авторизации
		2. Выбрать в меню "Зарегистрироваться"	Отображается форма регистрации
		3. Заполнить Поля: Поле почты ФИО	Поля успешно заполнены

		Пароль Подтверждение пароля Чекбоксы Кнопка регистрации	
		4. Нажать на кнопку Зарегистрироваться	Регистрация выполнена успешно
3.	Проверка формы авторизации UI	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается форма авторизации
		2. Проверка отображений меню для входа	Отображается Почта или логин Пароль Ссылка Забыли пароль Чекбокс Запомнить меня Вкладка Вход
4.	Проверка формы авторизации. Логин пользователя	1. Перейти на страницу http://study-test.local/my-account/	Страница успешно открыта, отображается форма авторизации
		2. Заполнить поля Имя пользователя	Имя успешно введено
		3. Заполнить поле Пароль	Пароль введён
		4. Клик на кнопку Войти	Вход успешно произведён

Ниже на рис. 1 представлена блок-схема работы предлагаемого метода многоаспектного скриншотного тестирования.

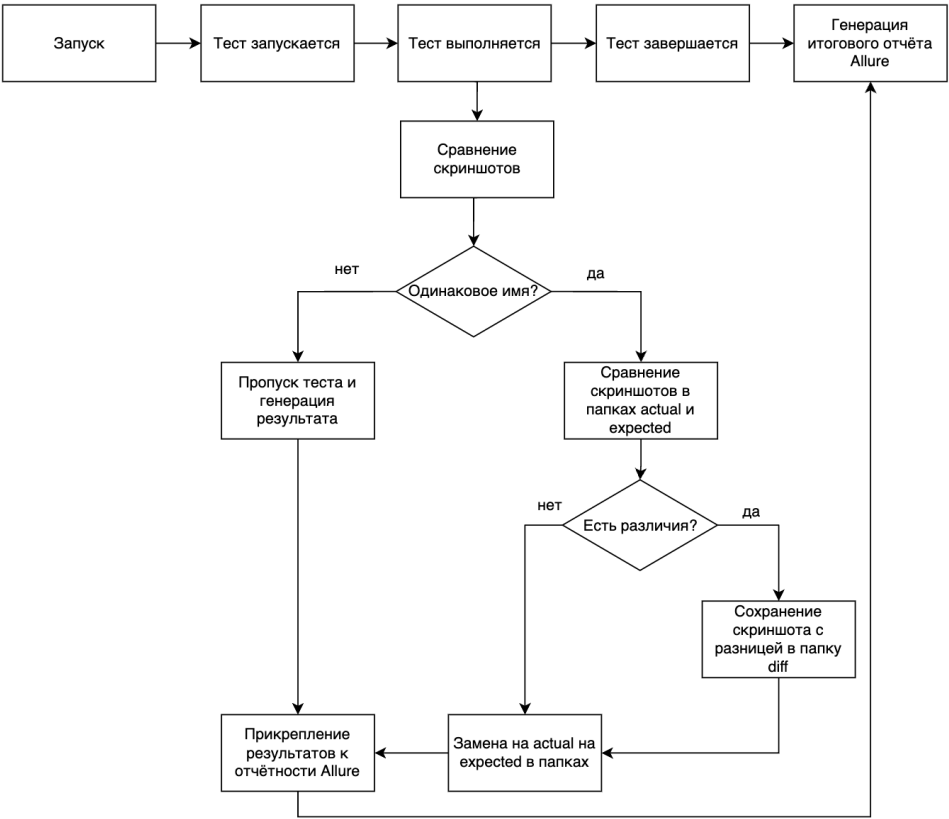


Рис. 1 — Блок схема выполнения тестов

Проверка страницы авторизации и регистрации

Шаг 1. Первый запуск тестирования проводился на эталонной версии ПО для оценки и отладки работоспособности созданного кода. В рамках этого шага все результаты должны быть pass. На рис. 2 приведены результаты тестирования эталонной версии ПО.

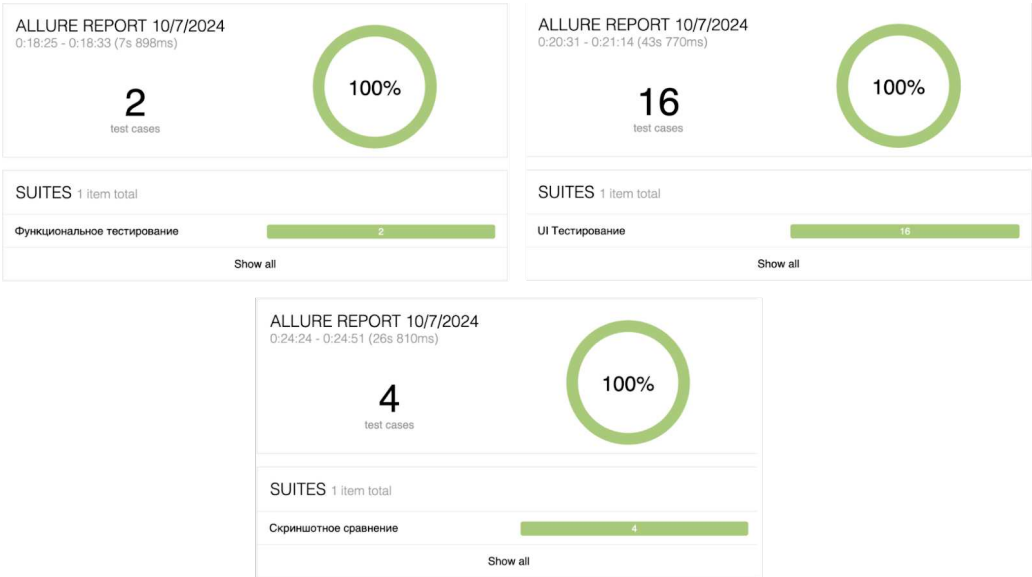


Рис. 2 — результаты функционального, UI и многоаспектного скриншотного тестирования эталонной версии ПО

Шаг 2. Для оценки результатов по выбранным метрикам необходимо внести изменения в тестируемую модель, чтобы оценить способность созданного решения выявлять ошибки. Изменения вносились в следующие аспекты: ширина полей формы авторизации изменена с 50 пикселей на 60 пикселей, белая заливка формы заменена с белой на

серую. Результаты показаны на рис. 3 и рис. 4.

Мой аккаунт

Войти

Зарегистрироваться

@

Email

First Name

Last Name

Password

Confirm Password

☐ I accept the [Terms of Service and Privacy Policy](#)

☐ Subscribe to our newsletter

Зарегистрироваться

Рис. 3 — визуальная демонстрация эталонной версии ПО

Мой аккаунт

Войти

Зарегистрироваться

@

Email

First Name

Last Name

Password

Confirm Password

☐ I accept the [Terms of Service and Privacy Policy](#)

☐ Subscribe to our newsletter

Зарегистрироваться

Рис. 4 — визуальная демонстрация версии ПО с имитацией ошибок

По результатам тестирования после внесения изменений (Рис. 5), имитирующих возникновение реальных ошибок на тестируемой модели, были получены следующие результаты:

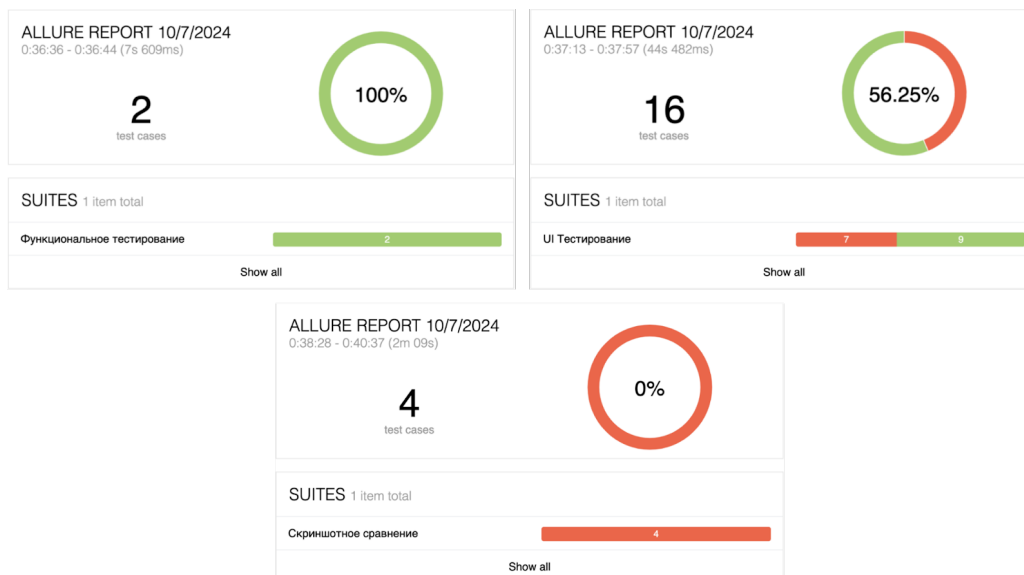


Рис. 5 — результаты функционального, UI и многоаспектного скриншотного тестирования версии ПО с внесенными ошибками

На рис. 5 показано, что функциональное тестирование не смогло полностью выполнить задачу верификации, так как его возможности не охватывают все аспекты, требующие проверки. В свою очередь, UI-тестирование выявило большее количество ошибок, что связано с более широким набором тестовых случаев, предусмотренных для проверки пользовательского интерфейса. Однако стоит отметить, что это тестирование не охватывало проверку функциональности, и поэтому не может служить полноценной заменой функциональному тестированию. В отличие от этого, многоаспектное скриншотное тестирование показало наилучший результат, выявив все ошибки и задокументировав их.

При нахождении ошибки многоаспектное скриншотное тестирование выводит в отчет сообщение, содержание которого зависит от проверяемого аспекта. Если ошибка была в визуальной составляющей, сообщение будет выглядеть следующим образом: «Failed: Ошибка при сравнении скриншотов: Разница найдена в скриншотах: 419178 пикселей отличаются от предыдущего скриншота: Проверка формы авторизации (UI). Наличие всех элементов_название файла.png», где «название файла.png» указывает на полученный после сравнения скриншот. Это упрощает процесс анализа и документации ошибок после проверки как для тестировщика, так и для разработчика.

Функциональное и UI тестирование на возникновение ошибки реагирует выводом статуса «AssertionError: <текстовое указание ошибки>», где <текстовое указание ошибки> может указывать, например «Не перешли на страницу успешной авторизации» или «Неверный color: rgba(160, 160, 160, 1)», что не столь очевидно и требует дополнительной работы для подготовки отчетности.

Результат наложения эталонной версии скриншота и той версии, которая снимается во время тестирования, показан на рис. 6. В таком виде после прохождения тестов формируются отчеты.

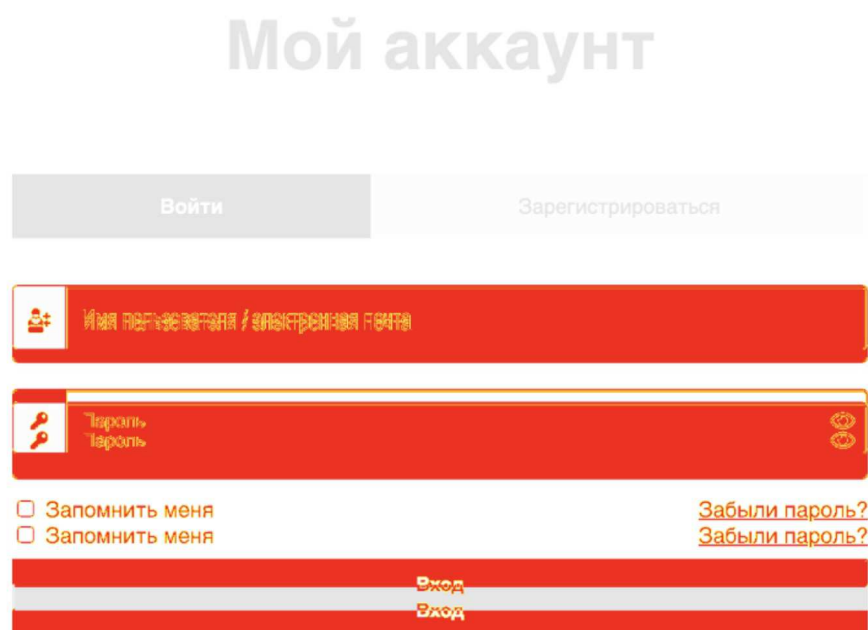


Рис. 6 — Пример итогового скриншота по результатам тестирования

По итогу проведенного эксперимента были получены следующие результаты:

Точность тестирования:

- **Функциональное тестирование:** выявлено 0 ложных срабатываний, пропущено 2 дефекта.
- **UI тестирование:** выявлено 7 дефектов, было 3 ложных срабатывания, пропущен 1 дефект.
- **Многоаспектное скриншотное тестирование:** выявлено 4 дефекта, 0 ложных срабатываний, пропущено 0 дефектов.

Для оценки точности проверяемых видов тестирования использовалась следующая формула:

$$x = \frac{y - z - f}{y + f} \times 100$$

Где x – точность, y – общее количество выявленных дефектов, z – количество ложных срабатываний, f – пропущенные дефекты.

Таким образом, получены следующие результаты:

- Точность функционального тестирования равна **100%**, однако на странице присутствовали UI дефекты, которые данным видом тестирования определяться не должны. С точки зрения поставленной задачи верификации функциональное тестирование покрывает только часть ошибок.
- Точность UI тестирования равна **37.5%**, так как было выявлено 3 ложных срабатывания и была пропущена 1 ошибка.
- Точность многоаспектного скриншотного тестирования равна **100%** так как были выявлены все дефекты без ложных срабатываний и пропущенных ошибок.

Этот расчёт подтверждает, что многоаспектное скриншотное тестирование выигрывает по

точности по сравнению с функциональным и UI тестированием, что делает его более эффективным методом для выявления дефектов в системе.

Стабильность тестовых сценариев:

Стабильность тестовых сценариев можно оценить на основе количества тестовых запусков, которые успешно проходят за несколько итераций, по сравнению с общим количеством тестовых запусков. Формула для расчёта стабильности использовалась следующая:

$$x = \frac{y}{z} \times 100$$

Где x – стабильность тестов, y – количество успешных тестовых запусков, z – общее количество тестовых запусков.

- **Успешные тестовые запуски** — это те тесты, которые прошли без сбоев или ошибок.
- **Общее количество тестовых запусков** — это общее число запусков тестов за несколько итераций.

Для получения данных по стабильности, тестовые запуски проводились несколько раз, после чего фиксировались результаты.

- **Успешные прогоны:** сколько раз тестовые запуски завершились успешно.
- **Неуспешные прогоны:** сколько раз тестовые запуски вышли из строя по причине непредсказуемых факторов (например, сбой в среде, проблемы с сетью).

Были получены следующие результаты:

1. Функциональное тестирование: 10 успешных запусков из 10. Стабильность равна 100%;
2. UI тестирование: 5 успешных запусков из 10, стабильность равна 50%;
3. Многоаспектное скриншотное тестирование: 9 успешных запусков из 10, стабильность равна 90%

Этот расчет показывает, что функциональное тестирование в рамках эксперимента показало наиболее стабильные результаты работы. Однако стоит принять во внимание то, что тестовые сценарии в рамках поставленной верификационной задачи были пройдены со значительными ошибками.

Сводную таблицу полученных результатов можно представить следующим образом:

Таблица 4

Полученные результаты в ходе эксперимента

Анализируемые положения для сравнения	Функциональное тестирование	UI тестирование	Скриншотное тестирование
Количество выявленных	0 из 2	7 из 16	4 из 4

дефектов			
Время выполнения тестов	11 секунд	45 секунд	28 секунд
Трудозатраты на разработку и поддержку тестов	Достаточно быстрые решения	Времязатратно, т.к. каждый аспект интерфейса требуется отдельно покрывать стилями и свойствами	Чуть дольше чем функциональное, но быстрее UI тестирования
Точность тестирования, включая ложные срабатывания и пропущенные дефекты	0%	37.5%	100%
Оценка стабильности тестов	100%	50%	90%

Таким образом, гипотеза подтверждается, многоаспектное скриншотное тестирование является более эффективным видом верификации веб-приложений по сравнению с функциональным и UI тестированием.

Обсуждение результатов

Предложенный метод проходил апробацию на искусственно созданном веб-приложении с ограничением проверки только критически важной составляющей в рамках четко сформулированной ограниченной задачи верификации. Предложенный метод верификации показал свою состоятельность, однако требует дополнительной проверки при интеграции в более сложные системы и постановки развернутой задачи верификации, которая будет требовать не только проверки критически важного функционала одной страницы, но и менее критичных, однако важных критериев верификации веб-приложения.

Стоит обратить внимание, что реализация многоаспектного скриншотного тестирования может быть сложной и требовательной к ресурсам задач. При увеличении количества проверяемых аспектов возрастает вероятность столкновения с проблемой экспоненциального роста сложности, что может привести к затруднениям в обработке всех возможных состояний системы. В контексте многоаспектного скриншотного тестирования это может проявляться в виде сложных интерфейсов с множеством динамически изменяющихся элементов, что требует ручной настройки системы оповещений.

В рамках предложенного метода мы обошли ограничение создав систему сравнения скриншотов и их последующего сохранения для отчетности. Схема такого решения

представлена выше на рис. 1. Каждый скриншот в хранилище именуется согласно определенной логике: в названии указывается дата выполнения теста, тестируемый функционал и аспект (например, это может быть тестирование одной из форм на странице и тогда скриншот будет содержать только ее, либо полная страница со всеми аспектами). В контексте регрессионного тестирования такой подход позволяет проводить регулярную проверку стабильности ПО, не затрачивая большого количества времени на прохождения однотипных тест-кейсов.

В случае работы с динамическими элементами, такими как автозамены номеров и т.д., описано решение, при котором данные элементы маскируются с помощью Javascript, что позволяет настроить автоматическую систему фильтрации для исключения их из проверки.

Также стоит подчеркнуть, что минусом многоаспектного скриншотного тестирования может являться его зависимость от вычислительных ресурсов. Поскольку метод предполагает одновременную проверку множества аспектов, это требует больших вычислительных мощностей, особенно при работе с крупными и сложными программами. Это может проявляться в необходимости хранения большого объема данных (скриншотов), а также в обработке изображений для их анализа. Это создает дополнительную нагрузку на систему и может увеличивать время выполнения тестов.

Данное ограничение можно обойти разными способами, в контексте данной работы используется компрессия изображений без потерь, которая позволяет значительно уменьшить объем сохраняемых данных. Таким образом, исходное изображение сохраняется без удаления какой-либо важной информации в формате PNG.

Также стоит учитывать, что для составления квартальной и годовой отчетности на многих проектах необходимо долговременное хранение данных. В тех случаях, когда отчеты после прохождения тестов уже обработаны и исправлены и быстрый доступ к ним не столь необходим, можно использовать методы архивации данных. Архивирование подразумевает объединение нескольких файлов в один файл с последующим сжатием для уменьшения общего объема данных. Например, при необходимости можно разбивать данные, полученные в ходе тестирования, по кварталам, в которых производились проверки, или по типу проверяемого функционала для упрощения управления файлами. Таким образом, в любой момент можно будет поднять именно тот пласт информации, который будет необходим. Архивация также является эффективным способом переноса данных для резервного копирования или передачи на удаленные серверы.

Так, реализация многоаспектного скриншотного тестирования может стать серьезной проблемой, если технические требования, процессы и методы, необходимые для конкретного ПО, не были проанализированы и описаны [\[9\]](#).

В контексте предложенного метода можно сформулировать следующие рекомендации для дальнейших исследований:

1. Провести дополнительные исследования по применению многоаспектного скриншотного тестирования для мобильных приложений, где влияние динамических элементов интерфейса и разнообразие устройств могут существенно усложнить процесс верификации;
2. Проверить метод многоаспектного скриншотного тестирования в рамках разных задач верификации с использованием более сложного веб-приложения, что потребует разработки алгоритмов оптимизации ресурсоемкости метода при работе с

крупными и высоконагруженными приложениями;

3. Исследовать возможности интеграции многоаспектного подхода с другими видами тестирования, а также усовершенствовать предложенный метод включением в него системы, которая имитирует поведение человека при тестировании программного обеспечения, используя последние достижения в области компьютерного зрения^[10].

Заключение

В рамках проведенного исследования была достигнута поставленная цель — доказана эффективность многоаспектного скриншотного тестирования как метода автоматизированной динамической верификации веб-приложений. На основе экспериментальных данных подтверждена гипотеза, что многоаспектное скриншотное тестирование превосходит традиционные методы, такие как функциональное и UI-тестирование, за счет гибридного анализа функциональных и визуальных характеристик приложения.

В ходе эксперимента, проведенного в рамках подготовки данной статьи, было показано, что возможности многоаспектного скриншотного тестирования превышают возможные ограничения и недостатки за счет сокращения времени на тестирование (особенно регрессионного), детализированного анализа разных аспектов, гибкости в изменении стратегии тестирования. Данные эксперимента могут быть экстраполированы на более сложные проектные решения, что делает применение многоаспектного скриншотного тестирования более эффективным способом динамической верификации ПО.

Предложенная методика базируется на теоретически обоснованных алгоритмах сравнения изображений, которые обеспечивают высокую точность анализа отклонений. Разработанная модель верификации позволяет эффективно структурировать тестовые сценарии и упрощает их анализ.

Таким образом, многоаспектное скриншотное тестирование позволяет создать гибкую и точную систему верификации, которая учитывает разные аспекты ПО за счет использования сразу нескольких видов тестирования в рамках одного запуска тестов, что позволяет автоматизировать сложные процессы с минимальным количеством ложных срабатываний. Более того, возможность сравнения между разными версиями приложения позволяет отслеживать изменения, что помогает убедиться в осознанном внесении правок, которые не нарушают функциональность или внешний вид ПО. Использование Python, Pytest и PIL дает возможность легко анализировать изменения между версиями и быстро выявлять неожиданные отклонения.

Библиография

1. The Economic Impacts of Inadequate Infrastructure for Software Testing. NIST Report, May 2002. Режим доступа: <https://www.nist.gov/system/files/documents/director/planning/report02-3.pdf> (Дата обращения: 05.10.2024).
2. Гурин Р. Е., Рудаков И. В., Ребриков А. В. Методы верификации программного обеспечения // Машиностроение и компьютерные технологии. 2015. № 10. С. 235-251.
3. Quadri S. M. K., Farooq S. U. Software testing-goals, principles, and limitations // International Journal of Computer Applications. 2010. Т. 6. № 9. С. 7-9.
4. Kumar S. Reviewing software testing models and optimization techniques: an analysis of efficiency and advancement needs // Journal of Computers, Mechanical and Management.

2023. Т. 2. № 1. С. 43-55.

5. Xie Q., Memon A. M. Designing and comparing automated test oracles for GUI-based software applications // ACM Transactions on Software Engineering and Methodology (TOSEM). 2007. Т. 16. № 1. С. 4.

6. Кудрявцева Е. Ю. Автоматизированное тестирование веб-интерфейсов // Горный информационно-аналитический бюллетень (научно-технический журнал). 2014. № S. С. 354-356.

7. Кулямин В. В. Методы верификации программного обеспечения / В. В. Кулямин. М.: ИСП РАН, 2008. 111 с.

8. Персиваль Г. Python. Разработка на основе тестирования / Г. Персиваль. М.: ДМК Пресс, 2018. 622 с.

9. Берегейко О. П., Дубовский А. С. Автоматизация тестирования веб-приложений // Вестник магистратуры. 2016. № 12-4 (63). С. 39-41.

10. Dwarakanath A., Neville D., Sanjay P. Machines that test Software like Humans. arXiv preprint arXiv:1809.09455 (2018).

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Представленная статья на тему «Скриншотное тестирование как многоаспектный вид автоматизированной динамической верификации веб-приложений» соответствует тематике журнала «Программные системы и вычислительные методы» и посвящена вопросу обеспечения высокого уровня качества и поддержания стабильного функционирования программного обеспечения. Так как даже малозначительные ошибки могут привести к потере стабильности, ухудшению пользовательского опыта и т. д., а как следствие к экономическим издержкам, важно регулярно проводить анализ состояния программного обеспечения.

В статье представлен широкий анализ литературных российских и зарубежных источников по теме исследования.

В качестве цели исследования авторы указывают разработку метода многоаспектного скриншотного тестирования, сочетающего функциональные и UI-подходы, для улучшения точности выявления дефектов и оптимизации ресурсов в процессе верификации веб-приложений.

Стиль и язык изложения материала является научным и доступным для широкого круга читателей. Статья по объему соответствует рекомендуемому объему от 12 000 знаков.

Статья достаточно структурирована - в наличии введение, заключение, внутреннее членение основной части (методы исследования, описание предлагаемого метода, экспериментальная апробация многоаспектного скриншотного тестирования, обсуждение результатов). В работе содержится графический материал, представленный 6 рисунками, а также 4 таблицы.

В качестве методологии исследования авторы указывают разработку и применение многоаспектного скриншотного тестирования, которое представляет собой комбинацию функционального и UI-тестирования, позволяющую эффективно решать задачи верификации веб-приложений. Основная идея метода заключается в проведении параллельного анализа функциональных и визуальных характеристик веб-приложения путем автоматического сравнения эталонных и тестовых скриншотов.

Также авторами проведен эксперимент, целью которого была проверка следующей гипотезы: многоаспектное скриншотное тестирование является более эффективным

видом верификации веб-приложений по сравнению с функциональным и UI тестированием, благодаря гибриднему анализу всех аспектов ПО в рамках выбранного тестового сценария за один запуск тестов. Оценка эффективности проводилась методом сравнительного анализа на примерах функционального, UI и многоаспектного скриншотного тестирования.

Практическая значимость статьи четко обоснована – доказана эффективность многоаспектного скриншотного тестирования как метода автоматизированной динамической верификации веб-приложений. На основе экспериментальных данных авторами подтверждена гипотеза, что многоаспектное скриншотное тестирование превосходит традиционные методы, такие как функциональное и UI-тестирование, за счет гибридного анализа функциональных и визуальных характеристик приложения.

Предложенный авторами метод проходил апробацию на искусственно созданном веб-приложении с ограничением проверки только критически важной составляющей в рамках четко сформулированной ограниченной задачи верификации. Предложенный метод верификации показал свою состоятельность, однако требует дополнительной проверки при интеграции в более сложные системы и постановки развернутой задачи верификации, которая будет требовать не только проверки критически важного функционала одной страницы, но и менее критичных, однако важных критериев верификации веб-приложения.

Статья «Скриншотное тестирование как многоаспектный вид автоматизированной динамической верификации веб-приложений» может быть рекомендована к публикации в журнале «Программные системы и вычислительные методы».

Программные системы и вычислительные методы

Правильная ссылка на статью:

Сергеев Д.А., Родионов Д.Г., Поляков П.А., Голиков Г.И., Старченкова О.Д., Дмитриев Н.Д., Конников Е.А.
Интеллектуальная система мониторинга и адаптации маршрута беспилотных летательных аппаратов на основе
нейросетевого анализа объектов риска // Программные системы и вычислительные методы. 2025. № 1. DOI:
10.7256/2454-0714.2025.1.73255 EDN: UZVYID URL: https://nbpublish.com/library_read_article.php?id=73255

Интеллектуальная система мониторинга и адаптации маршрута беспилотных летательных аппаратов на основе нейросетевого анализа объектов риска

Сергеев Дмитрий Анатольевич

кандидат экономических наук

доцент Высшей инженерно-экономической школы; Санкт-Петербургский политехнический университет Петра Великого

194021, Россия, г. Санкт-Петербург, ул. Новороссийская, 50

▫ sergeev_da@spbstu.ru



Родионов Дмитрий Григорьевич

ORCID: 0000-0002-1254-0464

доктор экономических наук

профессор; директор Высшей инженерно-экономической школы; Санкт-Петербургский политехнический университет Петра Великого

194021, Россия, г. Санкт-Петербург, ул. Новороссийская, 50

▫ drodionov@spbstu.ru



Поляков Прохор Александрович

исследователь лаборатории «Политех-Инвест»; Санкт-Петербургский политехнический университет Петра Великого

194021, Россия, г. Санкт-Петербург, ул. Новороссийская, 50

▫ prohor@polyakov-box.ru



Голиков Глеб Игоревич

ORCID: 0009-0008-0538-0942

исследователь лаборатории «Политех-Инвест»; Санкт-Петербургский политехнический университет Петра Великого

194021, Россия, г. Санкт-Петербург, ул. Новороссийская, 50

▫ golikov.23@mail.ru



Старченкова Олеся Дмитриевна

ORCID: 0009-0009-1168-2362

Специалист лаборатории «Моделирование и цифровизация социально-экономических систем»; Санкт-Петербургский политехнический университет Петра Великого

194021, Россия, г. Санкт-Петербург, ул. Новороссийская, 50

▫ starchenkova_od@spbstu.ru



Дмитриев Николай Дмитриевич

ORCID: 0000-0003-0282-1163

кандидат экономических наук

доцент; Высшая инженерно-экономическая школа; Санкт-Петербургский политехнический университет Петра Великого
Заведующий лабораторией; Лаборатория «Моделирование и цифровизация социально-экономических систем»; Санкт-Петербургский политехнический университет Петра Великого

194021, Россия, г. Санкт-Петербург, ул. Новороссийская, 50

✉ dmitriev_nd@spbstu.ru



Конников Евгений Александрович

ORCID: 0000-0002-4685-8569

кандидат экономических наук

доцент Высшей инженерно-экономической школы; Санкт-Петербургский политехнический университет Петра Великого
Заведующий лабораторией «Политех-Инвест»; Санкт-Петербургский политехнический университет Петра Великого

194021, Россия, г. Санкт-Петербург, ул. Новороссийская, 50

✉ konnikov_ea@spbstu.ru



[Статья из рубрики "Системный анализ, поиск, анализ и фильтрация информации"](#)

DOI:

10.7256/2454-0714.2025.1.73255

EDN:

UZVYID

Дата направления статьи в редакцию:

04-02-2025

Аннотация: Исследуется интеллектуальная система мониторинга и адаптации маршрута беспилотных летательных аппаратов (БПЛА) на основе нейросетевого анализа объектов риска. Рассматриваются алгоритмы автономной навигации, обеспечивающие анализ внешней среды и оперативную корректировку траектории полёта с учётом потенциальных угроз. Оцениваются возможности применения машинного зрения, нейросетевых алгоритмов, методов предобработки данных, детектирования объектов, семантической сегментации, алгоритмов траекторного планирования, предиктивного управления и адаптивной оптимизации маршрутов для идентификации препятствий, движущихся объектов и зон ограничения полётов. Анализируется роль интеллектуальных систем управления в архитектуре БПЛА, их влияние на повышение автономности, устойчивости и эффективности выполнения задач в динамически изменяющихся условиях. Предлагаемые решения ориентированы на снижение рисков, связанных с нештатными ситуациями, за счёт внедрения адаптивных стратегий управления полётом. Применяются методы системного анализа, компьютерного зрения и машинного обучения, включая

свёрточные нейросети, алгоритмы предобработки изображений, фильтрации и сегментации данных, а также анализ сенсорных показателей. Оценка эффективности реализована посредством моделирования траекторий движения, тестирования алгоритмов идентификации угроз и анализа параметров устойчивости маршрутов БПЛА. Научная новизна заключается в разработке интегрированной системы интеллектуальной корректировки маршрута БПЛА, основанной на применении нейросетевых методов классификации объектов и адаптивных алгоритмов планирования траекторий. Разработаны механизмы предиктивного анализа рисков, обеспечивающие автоматическую корректировку маршрута при обнаружении препятствий, неблагоприятных погодных условий и зон ограниченного доступа. Предложенная архитектура управления сочетает технологии машинного зрения, анализа потоков данных и автоматизированного принятия решений, а также использует методы динамической маршрутизации, алгоритмы корректировки полёта в реальном времени и стратегии предотвращения столкновений. Такой подход обеспечивает повышение уровня автономности работы дронов. Разработанные алгоритмы интеллектуальной навигации могут быть внедрены в современные системы автономного управления БПЛА, обеспечивая адаптацию к динамическим условиям и повышение эффективности выполнения задач в различных сферах, включая оборонные и промышленные применения.

Ключевые слова:

Система принятия решений, Интеллектуальное управление, Нейросетевой анализ, Машинное зрение, Глубокое обучение, Автономная навигация, Предотвращение угроз, Оптимизация маршрута, Беспилотные летательные аппараты, Адаптивные алгоритмы

Работы выполнены в рамках реализации проекта «Разработка методологии формирования инструментальной базы анализа и моделирования пространственного социально-экономического развития систем в условиях цифровизации с опорой на внутренние резервы» (FSEG-2023-0008).

Введение. С развитием технологий беспилотных летательных аппаратов (БПЛА) растет потребность в интеллектуальных системах управления, способных эффективно адаптировать маршрут с учетом реальных угроз и изменений окружающей среды [1]. Одним из перспективных направлений является использование нейросетевых технологий для анализа объектов риска и принятия оперативных решений о корректировке маршрута. В различных сферах, включая транспорт, промышленность, логистику и оборону, возрастает сложность задач, которые требуют автономного или полуавтономного управления [2]. В таких условиях традиционные системы управления уже не всегда способны эффективно справляться с динамическими изменениями окружающей среды. Это обусловлено несколькими ключевыми факторами: увеличение объема данных, сложные и динамичные условия эксплуатации, повышение требований к безопасности, спрос на оптимизацию ресурсов и необходимость автономной работы [3].

По уровню интеллекта выделяют следующие интеллектуальные системы управления [4]:

- Экспертные системы, которые используют заранее прописанные правила и базы знаний для выполнения задач.
- Обучаемые системы, в которых при помощи машинного обучения повышается

эффективность работы на основе новых данных.

· Системы с искусственным интеллектом представляют собой полностью автономные решения, способные анализировать данные, прогнозировать и принимать решения без вмешательства человека.

Цель статьи – создание устройства с искусственным интеллектом, которое в реальном времени:

1. Анализирует окружающую обстановку с использованием данных с камер, сенсоров и внешних источников.
2. Идентифицирует потенциальные угрозы (объекты риска), такие как погодные условия, препятствия, зоны ограничения полетов и движущиеся объекты.
3. Модифицирует маршрут БПЛА для обеспечения безопасности и повышения эффективности выполнения задачи.

Примером успешного выполнения задач в заявленном направлении являются изобретения российских ученых – новые модели дронов «Микроб» и «Касатка» с функциями искусственного интеллекта. Элементы ИИ позволяют дрону самостоятельно сопровождать и поражать цель [\[5\]](#):

1. Высокая степень автономности. Искусственный интеллект обеспечивает возможность функционирования дронов без непосредственного контроля оператора, что повышает их эффективность в сложных боевых условиях.
2. Возможность адаптации к изменяющейся обстановке в реальном времени. Он использует камеры и тепловизоры для построения карты местности и анализа объектов.
3. Распознавание дружественных и вражеских целей.
4. Возможность работы в составе роя дронов с распределением задач между аппаратами.

Особенность таких дронов – их системность, являясь частью единой концепции использования беспилотных систем, они взаимодействуют в рамках комплексной стратегии. «Касатка» анализирует данные о цели с помощью ИИ, принимает стратегические решения и координирует действия «Микробов» [\[6\]](#). Они, в свою очередь, выполняют тактические задачи, такие как сбор разведывательной информации, нанесение точечных ударов или отвлечение внимания противника. Применение дронов в рамках единой системы разведывательного обеспечения обеспечивает интеграцию данных с различных сенсоров и платформ, формируя целостную картину оперативной обстановки [\[7\]](#).

Данный аспект приводит авторов к ключевому термину «нейросетевой анализ объектов риска» – применению технологий искусственного интеллекта и глубокого обучения для выявления, классификации и прогнозирования потенциальных угроз в различных средах [\[8\]](#). Этот подход используется для анализа данных, поступающих от сенсоров, камер, радаров и других источников, и позволяет автоматизировать процесс оценки рисков, что особенно важно в динамичных и сложных условиях, таких как военные действия. Благодаря использованию данных с дронов, спутниковых снимков и наземных сенсоров нейросети могут быстро определять приоритетные цели, прогнозировать их поведение и предлагать оптимальные тактики реагирования. Такие технологии могут применяться для

оптимизации маршрутов войск или техники, минимизируя риски столкновения с опасностями, такими как артиллерийские обстрелы или зоны минных полей [\[9\]](#).

Реализация. В процессе анализа преимуществ внедрения был выделен комплекс преимуществ использования беспилотных летательных аппаратов для передачи информации. Среди таковых можно выделить [\[10\]](#):

1. Повышение оперативности развёртывания сетей связи. Использование БПЛА в качестве мобильных ретрансляторов позволяет существенно ускорить процесс развёртывания и перенастройки каналов связи на новом месте. При выходе из строя или недостаточной пропускной способности штатных узлов связи БПЛА могут оперативно заменить или дополнить существующую инфраструктуру, обеспечивая непрерывность командования и управления боевыми подразделениями [\[11\]](#).

2. Гибкая масштабируемость и резервирование. Благодаря возможности быстрого изменения количества и типа беспилотных летательных аппаратов, а также их распределения в зоне операции подобные комплексы позволяют динамически масштабировать сеть под конкретные задачи. При угрозе вывода из строя одного из звеньев сети (например, потери или технической неисправности одного дрона) другие БПЛА способны взять на себя его функции, обеспечивая высокий уровень отказоустойчивости [\[12\]](#).

3. Расширенный радиус действия и непрерывность связи. БПЛА могут действовать за пределами прямой видимости наземных узлов, увеличивая дальность и охват сети. Это особенно актуально в ситуациях, когда традиционные станции связи подвергаются сильным помехам или находятся в неблагоприятных условиях (горная местность, лесные массивы, инфраструктурные разрушения и т.д.) [\[13\]](#).

4. Эффективное противодействие радиоэлектронной борьбе. Высокая мобильность и быстрота перебазирования БПЛА осложняет их обнаружение и целенаправленное подавление средствами РЭБ [\[14\]](#).

5. Снижение нагрузки на спутниковые каналы. При высокой интенсивности обмена данными (например, в ходе крупной операции) спутниковые каналы могут перегружаться или быть ограничены помехами и внешними факторами. БПЛА, выступающие в роли альтернативных или дублирующих каналов связи, разгружают спутниковые сети и повышают общую пропускную способность комплексной системы связи [\[15\]](#).

6. Повышение конфиденциальности и трудность в обнаружении. Благодаря возможности изменения высоты полёта, маршрутов и дальности действия дроны могут создавать временные каналы связи, которые труднее отслеживать противнику. При условии передачи оперативных и разведывательных данных только посредством прямого физического контакта с дроном через USB-каналы, такое решение существенно затрудняет попытки перехвата информации [\[16\]](#).

Следовательно, с позиции технической реализации дрона в соответствии с изложенными теоретическими преимуществами критически важной является разработка системы автономной корректировки траектории движения БПЛА, что достигается посредством последовательного применения следующих программных и технических методов [\[17\]](#):

1. Система real-time видеофиксации окружения БПЛА, реализуемая посредством интеграции фото/видео аппаратуры в комплекс дополнительного навесного контура

- оборудования дрона [\[18\]](#).
2. Система машинного зрения, реализуемая нейросетевыми программными методами, которая ,в свою очередь, в целях обеспечения автономности операционной работы должна представлять собой интегрированное в БПЛА решение [\[19\]](#).
3. Система корректировки маршрута, оказывающая влияние на локальную краткосрочную траекторию движения БПЛА [\[20\]](#).

Комплексная техническая реализация таких решений достигается посредством разработки навесного контура. Принципиальная графическая электронная схема представлена на рисунке 1.

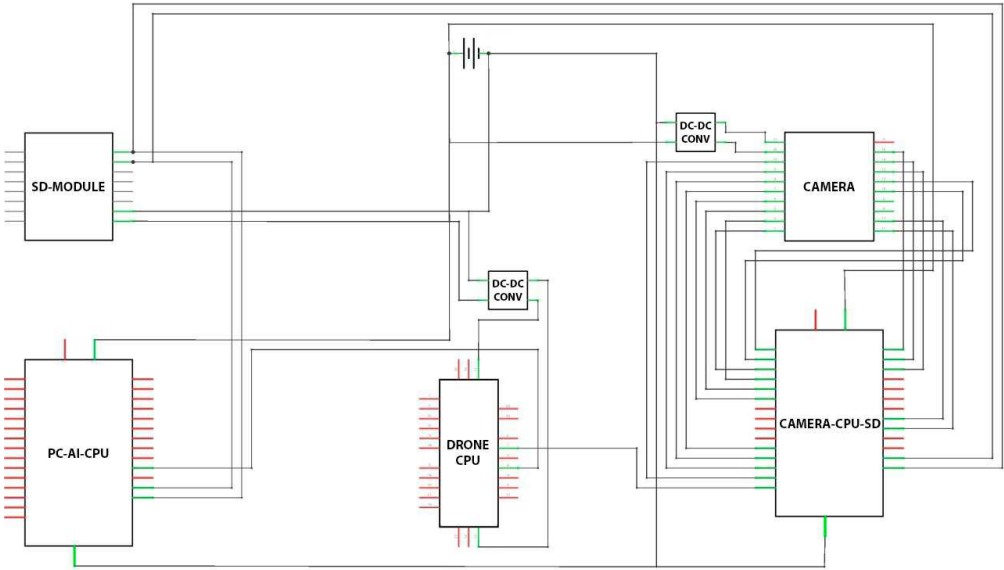


Рисунок 1 – Принципиальная графическая электронная схема навесного контура real-time корректировки маршрута

Элементы, представленные на рисунке 1, имеют следующее техническое и функциональное описание, рассмотренные в таблице 1. Логика их работы представлена в таблице 2.

Таблица 1 – Элементы в контуре системы и их описание

Компонент контура	Техническое и функциональное описание
Камера U2	Делает снимки в режиме реального времени, которые записываются на SD-модуль. Функционально большое количество подключений обусловлено необходимостью подачи питания, подключения параллельной передачи данных и синхронизации частоты съемки и времени передачи.
CAMERA-CPU-SD Raspberry Pi	Управляет камерой, регулирует её настройки и выполняет предварительную обработку снимков.
PC-AI-CPU Raspberry Pi	Использует полученные изображения для анализа с применением нейронной сети, выявления потенциальных угроз для дрона и выполнения поставленной задачи. В случае обнаружения угрозы

	подаёт обновлённую краткосрочную траекторию полёта на МК ESP.
DRONE-CPU ESP8266	Бортовой микроконтроллер дрона, интегрированный с ESC. Принимает от PC-AI-CPU скорректированные траектории полёта.
SD-MODULE	Модуль хранения SD-карты, на которую производится запись кадров с камеры и чтения снимков PC-AI-CPU Raspberry Pi.

Таблица 2 – Логика работы представленных компонентов в составе единого аппаратного контура

Шаг	Описание
1. Сбор данных	Камера фиксирует изображения с заданным временным интервалом и сохраняет их на SD-модуль.
2. Анализ данных	Raspberry Pi с помощью нейронной сети анализирует присутствие угроз для дрона на снимках. При обнаружении угроз формируется управляющий сигнал с формированием обновлённой краткосрочной траекторией полёта дрона.
3. Реакция на угрозы	ESP8266 – главный полётный микроконтроллер дрона, интегрированный с ESC. При необходимости он корректирует траекторию движения, принимая обновлённые параметры.
4. Управление и конфигурация	Предусмотрена возможность регулирования настроек камеры и обновления алгоритмов обработки изображений на втором Raspberry Pi.

Методология тестирования. Экспериментальная проверка была проведена с целью оценки работоспособности автономного беспилотного летательного аппарата при отсутствии GPS-сигнала. В состав системы входил модуль нейросетевого анализа угроз. Для тестирования были разработаны сценарии, имитирующие реальные условия эксплуатации. Испытания проводились как в симулированной среде, так и на открытой площадке, где GPS-сигнал был преднамеренно отключён или заглушен.

В качестве тестового устройства использовался квадрокоптер массой около ~1.5 кг, оснащённый бортовым компьютером для обработки нейросетевых алгоритмов. Навигационная система включала инерциальные датчики, барометр и монокулярную камеру.

Полёты выполнялись по заранее определённой маршруту длиной примерно ~500 м с возвратом в точку старта. В ходе экспериментов использовалась альтернативная навигация, основанная на данных одометров и алгоритмах компьютерного зрения. Для оценки корректности работы системы вводились искусственные угрозы. В качестве таковых выступали внезапно появляющиеся препятствия и запретные зоны, которые аппарат должен был идентифицировать с помощью обученной нейросети. Архитектура модели включала сверточную нейросеть, ориентированную на классификацию типовых угроз. В случае обнаружения опасности беспилотник должен был оперативно корректировать траекторию.

Для оценки эффективности проводился анализ реакции системы на различные факторы внешней среды. В ходе испытаний варьировались: скорость ветра (в диапазоне 0–8 м/с), плотность препятствий (редкие или многочисленные объекты), а также порог чувствительности детектора угроз. Каждая комбинация условий проверялась в не менее

чем 10 повторных полётах.

Фиксировались количественные параметры, отражающие успешность выполнения задач. В ходе тестирования анализировались: доля завершённых миссий; траектория движения аппарата; временной интервал между обнаружением угрозы и началом манёвра уклонения; количество ложных срабатываний детектора; точность классификации препятствий. Методологический подход позволил собрать статистически значимые данные и провести количественную оценку функционирования системы в воспроизводимых условиях.

Результаты экспериментов. На графике (Рис. 2) по оси X представлены изменяемые факторы, такие как скорость ветра (в м/с) и плотность препятствий (в баллах). По оси Y показан обобщённый критерий эффективности полёта, например, процент успешно пройденного маршрута или вероятность успешного завершения миссии.

Анализ данных показывает, что с увеличением сложности условий успешность выполнения полётов снижается монотонно. Увеличение скорости ветра свыше 5 м/с приводит к росту ошибки стабилизации и снижению числа успешно выполненных полётов. Ветровое воздействие до 2–3 м/с практически не влияет на результат.

Аналогичная тенденция прослеживается при изменении плотности препятствий. С увеличением числа объектов в зоне полёта возрастает время облёта и количество манёвров уклонения. После порога в 5 препятствий на 100 м пути наблюдается резкое снижение успешности миссии. В условиях экстремальной перегрузки маршрута показатель эффективности стремится к нулю.

Наибольшее влияние на успешность выполнения задач оказывает сложность окружающей среды. При высокой плотности препятствий и сильном ветре система достигает предельных значений устойчивости. Для повышения эффективности работы алгоритмов требуется адаптация методов навигации.

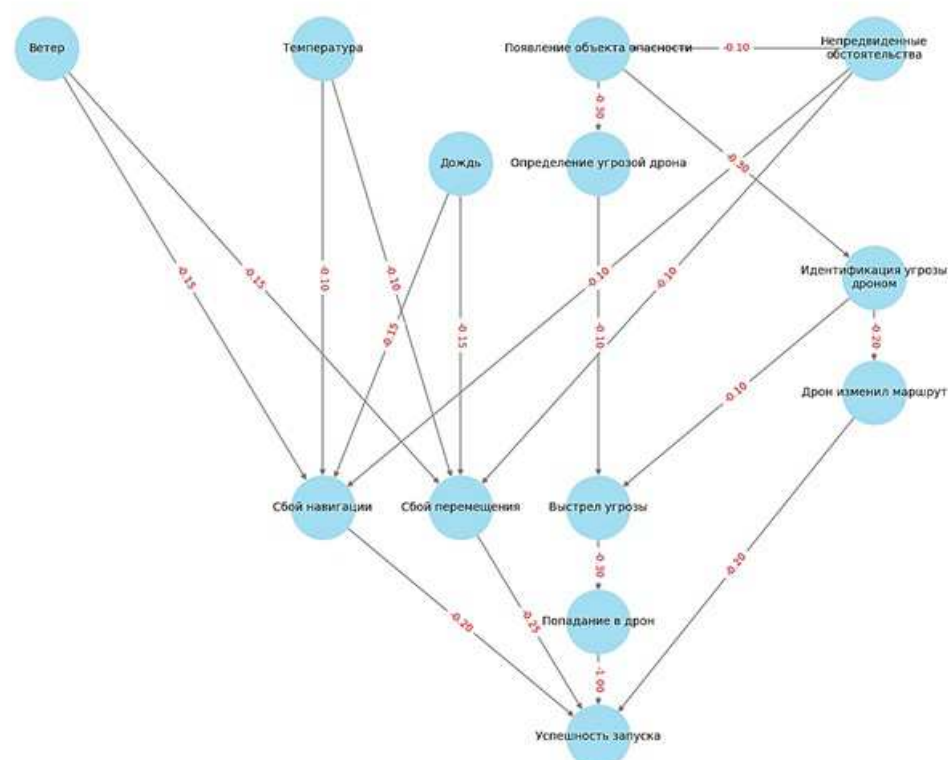


Рисунок 2 – График зависимости показателя успешности полёта от нескольких факторов среды демонстрирует чувствительность системы к внешним воздействиям.

На рисунке 3 представлена гистограмма плотности распределения вероятностей успешности полёта, отражающая статистическое распределение исходов по совокупности всех проведённых испытаний. По горизонтальной оси отложено значение рассчитанного показателя успешности полёта, а по вертикальной – плотность (частота) наблюдений данного значения. Распределение имеет выраженную двухмодальную форму. Основной пик сосредоточен в области положительных значений показателя (от 0 до 1), что соответствует успешным полётам. Меньший пик наблюдается в области отрицательных значений, что свидетельствует о частых отказах системы.

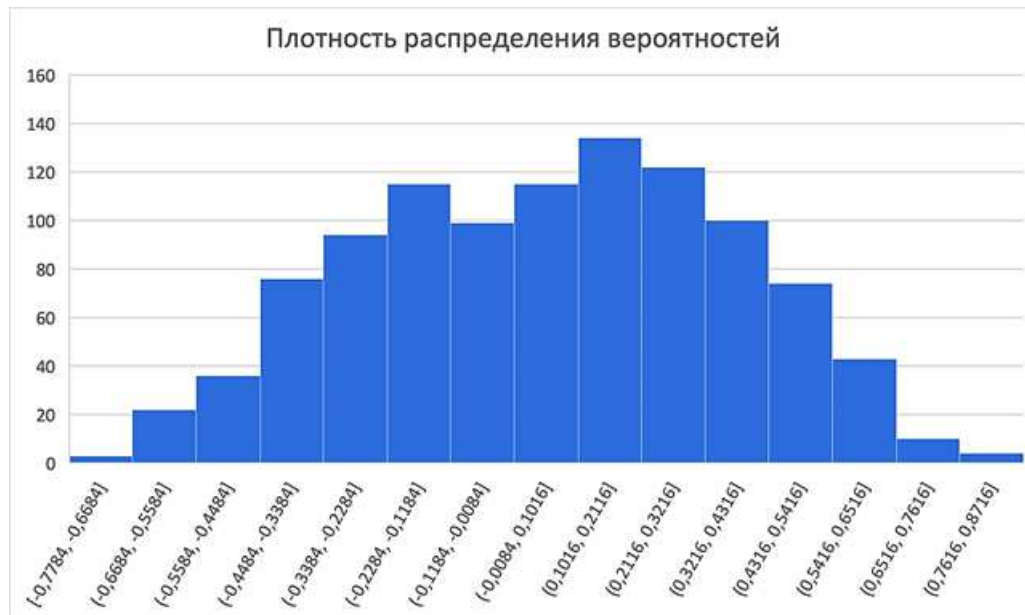


Рисунок 3 – Гистограмма плотности распределения вероятностей успеха полёта отражает статистическое распределение исходов по совокупности всех проведённых испытаний.

Анализ данных показывает, что у большинства запусков вероятность успешного завершения миссии находится в диапазоне 0.7–0.9, что подтверждает надёжность работы системы в штатных условиях. Однако зафиксирована группа испытаний, в которых система не смогла завершить миссию. Отрицательные значения по шкале X гистограммы (рисунок 3) соответствуют полному провалу полёта. В данных случаях дрон не достиг конечной точки или был вынужден выполнить аварийную посадку.

Площадь под левой частью гистограммы отражает частоту неудач. В ходе экспериментов около 15% запусков завершились отказом, тогда как 85% испытаний привели к успешному или частично успешному выполнению полётного задания.

Анализ результатов демонстрирует, что вероятность благоприятного исхода остаётся высокой в большинстве случаев. Однако сохраняется риск отказа системы, который необходимо учитывать при дальнейшей оптимизации алгоритмов и конфигурации аппаратного обеспечения.

Критический анализ. Полученные результаты подтверждают работоспособность предложенной системы, однако выявляют ряд ограничений текущей реализации.

Во-первых, зафиксирована зависимость эффективности от условий внешней среды. При сочетании сложных факторов система приближается к пределу своих возможностей, что свидетельствует о недостаточной робастности модели в экстремальных случаях.

Подобная уязвимость к нестандартным ситуациям характерна для автономной навигации. Динамичные и сложные условия создают значительные вызовы для беспилотного летательного аппарата [21].

Во-вторых, нейросетевой анализ угроз демонстрирует высокую точность в среднестатистическом случае, однако не является полностью безошибочным. Зафиксированы ложные срабатывания (false positives), когда безопасные объекты ошибочно классифицировались как угрозы, а также пропуски реальных препятствий. Подобные ошибки приводили либо к избыточно осторожным манёврам, либо к запоздалой реакции на угрозу. Сходные эффекты отмечены в других исследованиях. Например, показано, что даже при общей точности классификации около ~90% некоторые алгоритмы демонстрируют до 40% ложных тревог в сложных сценах [22]. В ходе проведённых испытаний ложные тревоги составили примерно ~8% обнаружений угроз. Данный результат свидетельствует о необходимости дальнейшей оптимизации алгоритмов фильтрации.

Дополнительным ограничением является накопление погрешности навигации при отсутствии GPS. Применение методов визуальной одометрии и инерциальных датчиков позволило снизить дрейф координат, однако ошибка продолжала накапливаться. На коротких дистанциях данное отклонение незначительно, но при длительном полёте без внешних ориентиров возможны значительные отклонения, которые в текущей реализации не компенсируются.

Кроме того, выявлено увеличение вычислительной нагрузки при активном использовании нейросетевого модуля. В ходе экспериментов зафиксировано возрастание среднего времени отклика системы примерно на 20%. Ограниченные ресурсы бортового компьютера и энергопотребление могут создать проблемы для обработки данных в режиме реального времени.

Наконец, испытания проводились в преимущественно контролируемых условиях. Не исследовались сценарии преднамеренного радиоэлектронного подавления каналов связи, а также появление принципиально новых типов угроз, отсутствующих в обучающей выборке нейросети.

Потенциальные направления оптимизации включают расширение и разнообразие обучающих данных для нейросети, что позволит снизить вероятность ошибок классификации. Перспективным решением является внедрение механизмов самокоррекции навигации, например, периодическая калибровка положения с использованием наземных ориентиров или альтернативных сенсоров. Существенного улучшения вычислительной эффективности можно достичь за счёт оптимизации программно-аппаратной части, что снизит задержки обработки данных. Дополнительные способы повышения устойчивости включают применение ансамблевых методов обнаружения угроз, совмещающих несколько алгоритмов, а также использование адаптивных стратегий планирования траектории с учётом степени уверенности нейросетевого детектора.

Количественные метрики и сравнение с существующими подходами. Для объективной оценки были рассчитаны количественные метрики, которые затем сопоставлялись с характеристиками альтернативных решений.

Средняя успешность полётов в условиях отсутствия GPS составила 85–88% (доля завершённых миссий без аварии). Данный результат несколько ниже, чем у аналогичных

систем, функционирующих при наличии устойчивого GPS-сигнала. Например, в одном из недавних исследований на симуляторе продвинутый метод на основе глубокого обучения продемонстрировал более 90% успешных полётов в сценарии с подавленным GPS. При использовании точных координат цели данный показатель достигал 95% [\[21\]](#). Средний результат текущей системы (~88%) близок к этому уровню, что подтверждает её конкурентоспособность.

При сравнении с традиционными алгоритмами навигации без GPS (например, инерциальная одометрия с реагированием на препятствия по заранее заданным правилам) зафиксировано явное улучшение. В ходе экспериментов упрощённая версия системы без нейросетевого модуля продемонстрировала успешность около ~70%. Более низкие показатели объясняются большим числом аварийных случаев из-за несвоевременного распознавания препятствий.

Одним из значимых параметров эффективности является время адаптации маршрута при возникновении угрозы. В представленном решении средний временной интервал между моментом обнаружения препятствия и началом манёвра уклонения составил порядка 1.3 секунд. Для сравнения, система без интеллектуального анализа угроз демонстрировала задержку до 2–3 секунд. Подобная разница объясняется проактивной природой нейросетевого детектора, который фиксирует препятствия раньше и тем самым сокращает время на перестроение траектории.

Точность классификации угроз в предложенном решении достигла 92% (по совокупности типов объектов и ситуаций). Данный показатель достаточно высок, однако уступает узкоспециализированным системам. Например, в задачах кибернетической безопасности беспилотников зафиксирована возможность достижения точности порядка ~99.99% на тестовых наборах данных [\[23\]](#). В условиях реального полёта с физическими угрозами достижение столь высокой точности затруднительно, поэтому 92% является удовлетворительным показателем.

Нейросетевой детектор демонстрирует баланс между точностью и широтой охвата угроз. Альтернативные системы, не использующие машинное обучение, обычно ориентированы либо на минимизацию ложных тревог ценой пропуска части опасных ситуаций, либо наоборот. Например, стандартные алгоритмы обнаружения препятствий (лидарные технологии или методы компьютерного зрения) обеспечивают приемлемую навигацию в простых условиях, но уступают представленному подходу по интегральной вероятности успешного полёта и скорости реагирования.

Сравнительный анализ метрик подтверждает, что внедрение нейросетевого анализа угроз в автономную навигацию беспилотного летательного аппарата без GPS даёт значительное преимущество в надёжности и эффективности полёта. Однако остаётся пространство для дальнейшей оптимизации модели и совершенствования алгоритмов.

Заключение. Разработанная система автономной корректировки траектории беспилотного летательного аппарата, основанная на нейросетевом анализе потенциальных угроз, демонстрирует эффективное взаимодействие аппаратных и программных модулей, обеспечивающих мониторинг и оперативное реагирование в реальном времени. Видеосенсорный блок осуществляет непрерывный сбор данных, которые анализируются алгоритмами машинного обучения для своевременного выявления опасных объектов и оценивания окружающей обстановки. Полученные данные применяются для формирования управляющих сигналов, обеспечивающих корректировку маршрута дрона и предотвращение столкновений или иных рисков.

Предложенная методика обеспечивает высокий уровень надёжности и автономности, что особенно значимо при функционировании в сложных и быстро изменяющихся условиях. Интеграция систем машинного зрения, анализа данных и принятия решений доказывает свою практическую ценность как в военных, так и в гражданских сценариях, где предъявляются жёсткие требования к точности, устойчивости связи и максимальной безопасности. Дальнейшая оптимизация таких решений способствует увеличению автономности беспилотных комплексов, повышает их эффективность в различных сферах и обеспечивает адаптацию маршрутов с учётом метеорологических условий, загруженности воздушного пространства и иных факторов, снижая материальные и человеческие потери.

Библиография

1. Яковлев А. Г., Баранов Д. Е., Шишкин М. Д. О современном состоянии и тенденциях применения новых материалов и технологий в конструкциях беспилотных летательных аппаратов // Вектор науки Тольяттинского государственного университета. 2014. № 1. С. 71-74.
2. Варламов О. О. Об одном подходе к метрике автономности и интеллектуальности робототехнических комплексов // Известия Кабардино-Балкарского научного центра РАН. 2017. № 6-2. С. 43-53.
3. Коптев В. А. Анализ БПЛА как актуальных радиолокационных целей и их маскирующих факторов // Международный журнал гуманитарных и естественных наук. 2024. № 6-1. С. 244-250.
4. Варламов О. О., Лазарев В. М., Чувиков Д. А., Джха П. О перспективах создания автономных интеллектуальных роботов на основе миварных технологий // Радиопромышленность. 2016. № 4. С. 96-105.
5. В России разработали новые многофункциональные дроны «Касатка» и «Микроб» [Электронный ресурс]. URL: <https://aif.ru/society/army/v-rossii-razrabotali-novye-mnogofunkcionalnye-drony-kasatka-i-mikrob> (дата обращения: 26.01.2025).
6. В России представили новые многофункциональные разработки для СВО [Электронный ресурс]. URL: <https://news.mail.ru/svo/64559772/> (дата обращения: 26.01.2025).
7. Малышев В. П., Виноградов О. В. Возможные направления повышения устойчивости организаций, необходимых для выживания населения при современных военных конфликтах // Проблемы анализа риска. 2023. № 5. С. 60-70.
8. Корнеев Д. С. Использование аппарата нейронных сетей для создания модели оценки и управления рисками предприятия // Управление большими системами: сборник трудов. 2007. № 17. С. 81-102.
9. Овчинский В. С. Технологии будущего против криминала. М.: ЛитРес, 2019.
10. Белякова Е. В., Прокопович Д. А., Рыжая А. А., Гильц Н. Е. Логистический подход к технологическому развитию промышленного комплекса региона // Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева. 2015. № 4. С. 1007-1013.
11. Вырелкин А. Д., Кучерявый А. Е. Использование беспилотных летательных аппаратов для решения задач «умного города» // Информационные технологии и телекоммуникации. 2017. № 1. С. 105-113.
12. Криничко В. А., Князев Е. А. Применение БПЛА в боевой работе командиров // Научный аспект. 2023. № 8. С. 537-546.
13. Довгаль В. А., Довгаль Д. В. Анализ безопасности роя дронов, противостоящего атакам злоумышленников // Дистанционные образовательные технологии: сборник трудов V Международной научно-практической конференции. Симферополь: Ариал, 2020. С. 372-377.

14. Ласточкин Ю. И., Ярыгин Ю. Н., Бывших Д. М. Методическое обеспечение обоснования способов боевого применения сил и средств радиоэлектронной борьбы при противодействии радиоэлектронной разведке в операциях объединений сухопутных войск // Военная мысль. 2018. № 6. С. 58-66.
15. Бурченков В. В. Автоматизированные системы контроля подвижного состава. Гомель: Белорусский государственный университет транспорта, 2020. 226 с.
16. Боровик В. С., Гуцул В. И., Клестов С. А., Фирсов И. С., Пыхтырев В. С., Шатравин В. В., Сырянкин В. И., Шидловский С. В., Шашев Д. В., Нгуен К. Т., Угрюмов Д. А., Сырянкин М. В. Коллективы интеллектуальных роботов. Сферы применения. Томск: СТТ, 2018. 140 с.
17. Терентьев В. Б. Метод свёртки вероятностных и параметрических показателей летательного аппарата // Фундаментальные и прикладные научные исследования: актуальные вопросы. 2019. С. 51.
18. Сорокин И. А., Романов П. Н., Чесноков А. Д., Кондраненкова Т. Е. Математическая модель обработки изображений опасных и вредоносных растений с камеры БПЛА // Вестник НГИЭИ. 2023. № 5. С. 20-32.
19. Али Б., Садеков Р. Н., Цодокова В. В. Алгоритмы навигации беспилотных летательных аппаратов с использованием систем технического зрения // Гироскопия и навигация. 2022. № 4. С. 87-105.
20. Костин А. С. Информационно-измерительные системы для контроля выполнения траектории авиационной системы // Волновая электроника и инфокоммуникационные системы: материалы XXIV Международной научной конференции. Часть 1. 2021. С. 219-226.
21. Xu Y., Wei Y., Wang D., Jiang K., Deng H. Multi-UAV Path Planning in GPS and Communication Denial Environment // Sensors (Basel). 2023. Vol. 23. № 6. P. 2997.
22. Vera-Yanez D., Pereira A., Rodrigues N., Molina J.P., García A.S., Fernández-Caballero A. Vision-Based Flying Obstacle Detection for Avoiding Midair Collisions: A Systematic Review // J. Imaging. 2023. Vol. 9. P. 194.
23. Alharthi R. Enhancing unmanned aerial vehicle and smart grid communication security using a ConvLSTM model for intrusion detection // Front. Energy Res. 2024. Vol. 12. P. 1491332.

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Статья посвящена разработке интеллектуальной системы мониторинга и адаптации маршрута беспилотных летательных аппаратов (БПЛА) с применением технологий искусственного интеллекта и нейросетевого анализа. Основное внимание уделено методам идентификации потенциальных угроз в режиме реального времени и корректировке траектории дронов для обеспечения их безопасности и повышения эффективности выполнения задач.

Авторы опираются на комплексный подход, включающий:

- Анализ существующих интеллектуальных систем управления БПЛА.
- Описание принципов работы экспертных систем, машинного обучения и ИИ в контексте автономного управления.
- Разработку модели автономной корректировки маршрута с применением машинного зрения и анализа данных с сенсоров.
- Практическую реализацию системы на базе Raspberry Pi и ESP8266 с применением

камер, модулей хранения данных и программных алгоритмов машинного обучения.

Использованный методологический подход логичен и обоснован, однако в статье отсутствует экспериментальная проверка работоспособности системы на реальных данных или ее сравнение с существующими решениями. Добавление таких аспектов значительно повысило бы доказательность представленных результатов.

Работа затрагивает актуальные вопросы автономного управления БПЛА, которые находят применение в различных сферах – от военных операций до логистики и промышленности. Учитывая стремительный рост использования беспилотных систем, тема является востребованной. Особенно важен предложенный подход к анализу рисков и модификации маршрутов на основе данных с камер и сенсоров.

Научная новизна статьи заключается в предложенной системе нейросетевого анализа рисков и адаптации маршрута БПЛА в реальном времени. Основные нововведения включают:

- Применение машинного зрения для идентификации угроз.
- Автоматическую корректировку траектории с учетом обнаруженных факторов.
- Интеграцию нескольких аппаратных и программных компонентов в единую систему.

Тем не менее, статья могла бы включать более детальное сравнение с существующими методами, что позволило бы точнее оценить новизну подхода.

Статья написана в академическом стиле, структура работы логична и последовательна. Выделены основные разделы, такие как введение, описание методов, реализация и заключение. Однако текст содержит избыточные описания технических деталей аппаратных решений, что усложняет восприятие информации. Было бы полезно сократить или вынести технические подробности в отдельные приложения.

В статье практически отсутствует критическое обсуждение альтернативных подходов и возможных недостатков предложенной системы. Было бы полезно включить анализ ограничений работы, возможных направлений улучшения модели и сравнение с существующими решениями.

Выводы четко обобщают результаты исследования и подтверждают эффективность предложенной системы. Однако они носят преимущественно описательный характер и не включают количественные метрики, подтверждающие практическую ценность разработанной модели. Следует добавить сравнительный анализ с другими системами или результаты тестирования.

Работа представляет интерес для специалистов в области автоматизированных систем управления, искусственного интеллекта и разработки БПЛА. Ее могут использовать как исследователи, так и инженеры, занимающиеся разработкой автономных дронов.

Рекомендации по доработке:

1. Добавить экспериментальную проверку системы – представить данные тестирования, оценку точности обнаружения угроз, сравнение с альтернативными методами.
2. Развить критический анализ – включить обсуждение возможных недостатков метода, ограничений работы и направлений дальнейших исследований.
3. Уточнить выводы – добавить количественные метрики, подтверждающие эффективность предложенной модели.

Статья представляет собой интересное исследование в области интеллектуальных систем управления БПЛА. Работа актуальна, но требует доработки в части экспериментального подтверждения результатов, оформления и критического анализа. При исправлении указанных замечаний статья может быть рекомендована к публикации.

Рекомендация: доработка статьи с последующей повторной подачей.

Результаты процедуры повторного рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Представленная статья на тему «Интеллектуальная система мониторинга и адаптации маршрута беспилотных летательных аппаратов на основе нейросетевого анализа объектов риска» соответствует тематике журнала «Программные системы и вычислительные методы» и посвящена актуальному вопросу использования нейросетевых технологий для анализа объектов риска и принятия оперативных решений о корректировке маршрута, что вызвано развитием технологий беспилотных летательных аппаратов и потребностью в интеллектуальных системах управления, способных эффективно адаптировать маршрут с учетом реальных угроз и изменений окружающей среды.

В статье представлен достаточно широкий анализ литературных российских и зарубежных источников по теме исследования.

В качестве цели исследования авторы указывают создание устройства с искусственным интеллектом, которое в реальном времени:

1. Анализирует окружающую обстановку с использованием данных с камер, сенсоров и внешних источников.
2. Идентифицирует потенциальные угрозы (объекты риска), такие как погодные условия, препятствия, зоны ограничения полетов и движущиеся объекты.
3. Модифицирует маршрут БПЛА для обеспечения безопасности и повышения эффективности выполнения задачи.

Стиль и язык изложения материала является научным и доступным для широкого круга читателей. Статья по объему соответствует рекомендуемому объему от 12 000 знаков.

Статья достаточно структурирована - в наличии введение, заключение, внутреннее членение основной части (реализация, методология тестирования, результаты экспериментов, критический анализ, количественные метрики и сравнение с существующими подходами).

Авторами проанализированы преимущества использования беспилотных летательных аппаратов для передачи информации. А также определено последовательное применение программных и технических методов, необходимых для разработки системы автономной корректировки траектории движения БПЛА с позиции технической реализации дрона. В качестве решения данной задачи авторы предлагают комплексную техническую реализацию посредством разработки навесного контура. Принципиальная графическая электронная схема представлена на рисунке. Также авторы дают техническое и функциональное описание элементов, представленных на графической электронной схеме, результаты представлены в таблице. Логика работы представленных компонентов в составе единого аппаратного контура также представлена авторами в табличном виде. Была проведена экспериментальная проверка с целью оценки работоспособности автономного беспилотного летательного аппарата при отсутствии GPS-сигнала. Фиксировались количественные параметры, отражающие успешность выполнения задач. В ходе тестирования анализировались: доля завершённых миссий; траектория движения аппарата; временной интервал между обнаружением угрозы и началом манёвра уклонения; количество ложных срабатываний детектора; точность классификации препятствий. Методологический подход позволил собрать статистически значимые данные и провести количественную оценку функционирования системы в воспроизводимых условиях. Результаты экспериментов представлены в графическом виде. Полученные результаты подтверждают работоспособность предложенной системы, однако авторами выявлен ряд ограничений текущей реализации - зависимость

эффективности от условий внешней среды; высокая точность в среднестатистическом случае, однако не является полностью безошибочным; накопление погрешности навигации при отсутствии GPS; увеличение вычислительной нагрузки при активном использовании нейросетевого модуля.

Для объективной оценки авторами рассчитаны количественные метрики, которые затем сопоставлялись с характеристиками альтернативных решений.

Практическая значимость статьи четко обоснована. Разработанная система автономной корректировки траектории беспилотного летательного аппарата, основанная на нейросетевом анализе потенциальных угроз, демонстрирует эффективное взаимодействие аппаратных и программных модулей, обеспечивающих мониторинг и оперативное реагирование в реальном времени.

Статья «Интеллектуальная система мониторинга и адаптации маршрута беспилотных летательных аппаратов на основе нейросетевого анализа объектов риска» может быть рекомендована к публикации в журнале «Программные системы и вычислительные методы».

Программные системы и вычислительные методы

Правильная ссылка на статью:

Савельева Ю.О., Ильин А.В., Лукьянов А.А., Милихина К.М. Разработка прикладного приложения для автоматизации проектирования в КОМПАС-3D: таблица контроля качества основного металла // Программные системы и вычислительные методы. 2025. № 1. DOI: 10.7256/2454-0714.2025.1.73073 EDN: VBHBRP URL: https://nbpublish.com/library_read_article.php?id=73073

Разработка прикладного приложения для автоматизации проектирования в КОМПАС-3D: таблица контроля качества основного металла

Савельева Юлия Олеговна

ORCID: 0009-0004-8971-4110

кандидат технических наук

доцент, Филиал ФГБОУ ВО "СамГТУ" в г. Сызрани; Самарский государственный технический университет

446001, Россия, Самарская область, г. Сызрань, ул. Советская, 45, каб. 404

✉ Savelieva_yu_ol@mail.ru



Ильин Артем Владимирович

студент, Филиал ФГБОУ ВО "СамГТУ" в г. Сызрани; Самарский государственный технический университет

446001, Россия, Самарская область, г. Сызрань, ул. Советская, 45

✉ Savelieva_yu_ol@mail.ru



Лукьянов Александр Александрович

студент, Филиал ФГБОУ ВО "СамГТУ" в г. Сызрани; Самарский государственный технический университет

446001, Россия, Самарская область, г. Сызрань, ул. Советская, 45

✉ Savelieva_yu_ol@mail.ru



Милихина Ксения Михайловна

студент, Филиал ФГБОУ ВО "СамГТУ" в г. Сызрани; Самарский государственный технический университет

446001, Россия, Самарская область, г. Сызрань, ул. Советская, 45

✉ Savelieva_yu_ol@mail.ru



[Статья из рубрики "Системный анализ, поиск, анализ и фильтрация информации"](#)

DOI:

10.7256/2454-0714.2025.1.73073

EDN:

VBHBRP

Дата направления статьи в редакцию:

18-01-2025

Аннотация: Сложность составления таблиц контроля качества основного металла ТБ1 для проектирования изделий атомной энергетики определяется необходимостью учета совокупности множества требований различных источников (которые могут пересекаться, дополнять друг друга): стандарты, например, отраслевой стандарт (ОСТ), строительные нормы и правила (СНиП), технические условия, требования заказчика, а также внутренние нормы на их заполнение и другие нормативные документы. Некорректно составленные документы, переданные в уполномоченную организацию контроля качества (организацию уполномоченную на проведение работ по оценке соответствия в форме приемки (испытаний) продукции, предназначенной для применения в элементах объектов использования атомной энергии Российской Федерации), означают для разработчика финансовые и временные потери, трудности при дальнейшей сдаче изделия. В данной работе приводится порядок разработки прикладного пользовательского приложения (библиотеки), позволяющего автоматически сформировать и построить таблицу контроля качества основного металла ТБ1 на документе чертежа в КОМПАС-3D. Разработан код прикладного приложения (библиотеки) для САПР КОМПАС-3D на языке Delphi, с помощью которого реализуется автоматизированное построение ТБ1, необходимой при проектировании оборудования для атомной энергетики. Одной из важнейших характеристик современной инженерной системы моделирования является возможность внедрения автоматизации, например, использование подсистем для расширения возможностей программного пакета, в качестве которых могут выступать прикладные приложения (пользовательские программные библиотеки). В данной работе сформулированы основные требования к функционалу прикладного пользовательского приложения (библиотеки) для КОМПАС-3D: детали осуществления запуска формирования ТБ1; возможности выбора контрольных операций для материалов сборки, операций с шифрами, также добавление новых операций, с последующим их сохранением в памяти программы; возможность внесения оперативных изменений. Код приложения позволяет считать данные с файла сборки, открытого в КОМПАС-3D, создать новый документ чертежа, построить таблицу по требованиям ТБ1, занести туда данные, взятые со сборочного документа.

Ключевые слова:

таблица контроля качества, основной металл, Т Б 1 , КОМПАС-3D, прикладное приложение, пользовательская библиотека, программная библиотека, автоматизация проектирования, Delphi, инструменты создания библиотек

Введение. Эффективность программного пакета КОМПАС-3D для трехмерного моделирования, создания чертежей, спецификаций определяется использованием не только базовых инструментальных средств для реализации целевой функции, но и возможностью расширения функционала программного пакета с помощью прикладных

пользовательских приложений, пользовательских программных библиотек [\[1-5\]](#). В САПР КОМПАС-3D уже предусмотрено такое средство автоматизации, как встроенные прикладные библиотеки – подключаемые программные модули, число которых в разных версиях отличается, позволяющие сократить время проектирования за счет простоты и высокой скорости создания и использования стандартных элементов на основе базовых функций [\[6-8\]](#).

Несмотря на значительное количество библиотек (конструкторской библиотеки, содержащей более двухсот параметрических изображений типовых элементов: различных крепежных элементов, подшипников, профилей, элементов соединений трубопроводов и др.; библиотеки редукторов, электродвигателей, крепежей, канавок, пружин, деталей штампов, пресс-форм, материалов и сортаментов и прочих [\[6\]](#)), не всегда их достаточно для реализации целевой функции. Например, какой-то конкретной библиотеки нет в свободном доступе или на предприятиях разрабатываются изделия по собственным нормативным документам. Готовые модули для разработки приложений в КОМПАС содержат в себе описание интерфейсов в соответствии с синтаксисом определенного языка программирования.

На предприятиях при проектировании и изготовлении оборудования атомной энергетики составляются таблицы контроля качества основного металла (ТБ1), на базе которых разрабатываются программы качества. Таблицам контроля качества, устанавливающим необходимость выполнения конкретных контрольных операций, присваиваются определенные шифры: «ТБ1» - таблицы контроля качества основного металла; «ТБ2» - таблицы контроля качества сварных соединений и наплавов; «ТБ3» - таблицы контроля качества производственных сварных соединений на производственных контрольных сварных соединениях [\[9\]](#).

Проекты с использованием ТБ1 проходят приёмку в уполномоченной организации по программам контроля качества [\[10-11\]](#). Программа контроля качества изделий атомной энергетики, состоящая из текстовой части и таблиц контроля качества, определяет принципиальные положения и требования, предъявляемые к качеству проектируемых сборочных единиц, изделий, оборудования и трубопроводов, установок в процессе изготовления [\[9\]](#).

Составление ТБ1 – сложный процесс, т.к. конструктору-исполнителю при проектировании необходимо опираться на различные стандарты, например, на отраслевые стандарты (ОСТ), государственные стандарты, строительные нормы и правила (СНиП), технические условия и другие нормативные документы, действуя в соответствии с требованиями главного проектировщика – заказчика. Некорректно составленные документы, переданные в уполномоченную организацию контроля качества, влекут за собой финансовые и временные потери для разработчиков, а также трудности при дальнейшей сдаче изделия.

Объект исследования: таблица контроля качества основного металла (ТБ1).

Предмет исследования: порядок разработки прикладного пользовательского приложения.

Постановка задачи. Необходимо разработать прикладное пользовательское приложение для Компас-3D позволяющее автоматически сформировать ТБ1.

Основные требования к функционалу пользовательского приложения:

1. Запуск происходит со сборки в КОМПАС-3D через панель команд или через кнопки «Приложения» - «ТБ1».
2. После запуска в окне программы отображается таблица с полученными со сборки деталями (также детали всех подсборок), их количеством, обозначением, материалом и соответствующим стандартом.
3. В программе должен быть функционал позволяющий конструктору менять все графы таблицы, выбирать те позиции, которые построятся в таблице на чертеже, добавлять новые позиции, удалять добавленные, зачеркивать текст и возможность разделить таблицу на несколько листов.
4. В приложении должен быть предусмотрен выбор контрольных операций для материалов сборки, операций с шифрами, также добавление новых операций, с последующим их сохранением в памяти программы.
5. Возможность выбора символов для операций по клику на соответствующую ячейку.
6. По нажатию на соответствующую кнопку автоматически должен создаваться документ чертежа, с построенной таблицей со всеми изменениями, внесенными конструктором в интерфейс программы.
7. Возможность добавить функционал считывания данных с чертежа для внесения изменений через интерфейс программы.

В данной работе построение ТБ1 осуществляется согласно ОСТ 108.004.10-86^[9] и внутренним нормам на их заполнение у заказчика. Программный код приложения написан на языке Delphi.

Порядок разработки прикладного пользовательского приложения. Создается «заготовка» для приложения (библиотеки) КОМПАС:

- 1) В Delphi выполняется команда File - New - Other, в появившемся окне New Items со всеми возможными шаблонами Windows-приложений и файлами Delphi (исполняемый exe-файл, консольное приложение, форма или компонент Delphi и пр.) на вкладке «Delphi Projects» выбирается пункт Dynamic-Link Library.
- 2) Подключаются файлы, которые содержат в себе описание всех интерфейсов API, версии 7. Для этого выполняется команда Tools - Options – Library, в графе Library path указывается путь к каталогу include, который находится в папке КОМПАС.
- 3) Выполняется команда Projects - options - Application. В поле Target file extension вводится расширение.
- 4) В проект необходимо добавляется новый модуль и 64-битная платформа, выполняется сохранение проекта и компиляция.
- 5) Чтобы «заготовка» могла взаимодействовать с КОМПАС, в ней должны присутствовать функции: LibraryEntry (точка входа в библиотеку), LibraryId (возвращает идентификатор библиотеки), GetNewKompasAPI (получение доступа к новому API; не обязательна, но удобна тем, что не приходится каждый раз писать проверку на API). Эти функции обязательно должны быть экспортными, т.е. экспортируемыми, чтобы система КОМПАС могла их видеть и вызывать; по этой причине они выносятся в раздел exports.

Необходимо осуществить считывание данных со сборки. Все основные процедуры и

функции взаимодействия приложения с КОМПАС-3D находятся в модуле UstatementMaterials.

Как и во всех библиотеках под КОМПАС-3D используется процедура LIBRARYENTRY, код показан на Рис. 1.

```

procedure LIBRARYENTRY( command: WORD ); pascal;
var
  handleFAssembly, handleFDrawing, HandleFActiveList: HWND;
begin
  //доработанная процедура LIBRARYENTRY
  handleFAssembly:= FindWindow(PWideChar('TFmain'),'ТБ1 (со сборки)');
  //поиск окна с названием программы
  handleFDrawing:= FindWindow(PWideChar('TFmain'),'ТБ1 (с чертежа)');
  HandleFActiveList:= FindWindow(PWideChar('TFmain'),'ТБ1 (Активный лист)');
  //если найдено окно с таким же названием вывод сообщения
  if ((handleFAssembly <> 0) or (handleFDrawing <> 0) or (HandleFActiveList
<> 0)) then
  begin
    ShowKompasMessage('Приложение уже запущено.', );
    if handleFAssembly <> 0 then
    begin
      SetFocus(handleFAssembly);
    end
    else if handleFDrawing <> 0 then
    begin
      SetFocus(handleFDrawing);
    end
    else if HandleFActiveList <> 0 then
    begin
      SetFocus(HandleFActiveList);
    end;
    Exit;
  end;

  Application.Handle := GetHWND;
  GetNewKompasAPI;

  case command of
    1: CreateLineDimension;
  end;
end;

```

Рис. 1. Процедура LIBRARYENTRY

Процедура модуля CreateLineDimension, с помощью которого проверяется активный документ (если запущена сборка (Рис. 2), то данные считаются со сборки (в качестве примера тестовой сборки приводится термопистолет), если чертеж – с чертежа), приводится на Рис. 3. Вызываются функции, с помощью которых считываются данные со сборки и с чертежа, GetAssemblyComponents и GetDrawingComponents (Рис. 4), все полученные данные заносятся в ComponentsList (Рис. 5).

С помощью процедуры FormCreate создается форма и ее компоненты (Рис. 6 и Рис. 7), из ComponentsList данные заносятся в интерфейс приложения компонент TreeList (Рис. 8).

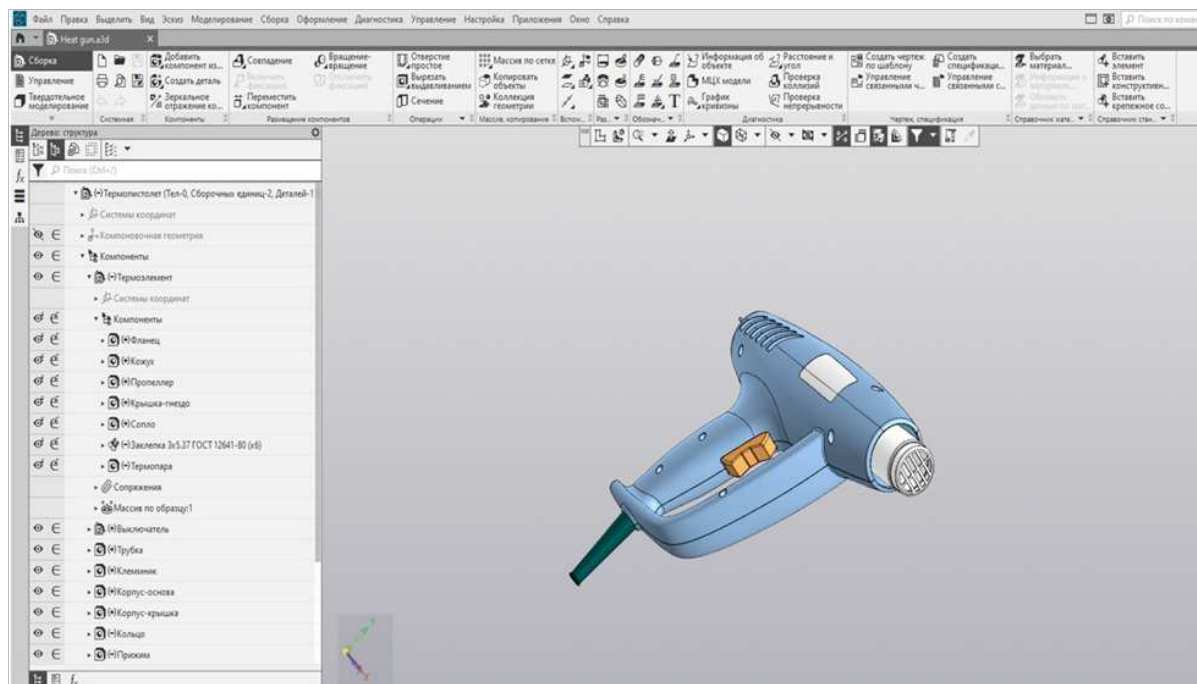


Рис. 2. Тестовая сборка

```

procedure CreateLineDimension;
var
  SR: TSearchRec;
begin
  GetProgramPath; //Получения пути до программы

  //Для загрузки операций и настроек
  if FindFirst(ProgramPath, faDirectory, SR) <> 0 then
  begin
    ShowKompasMessage('Не найден корневой каталог. Требуется переустановка
программы.', 2);
    exit;
  end;
  kDoc := newKompasAPI.ActiveDocument;
  if (kDoc <> nil) then if (kDoc.DocumentType = ksDocumentAssembly) or
(kDoc.DocumentType = ksDocumentDrawing) then
  begin
    GetSettingsVariables; //получение настроек с файла
    ReportMemoryLeaksOnShutdown := true;
    //листы для хранения данных
    OperList:= TList<TOper>.create; //операции
    ComponentsList:= TList<TComponents>.create; //запись с компаса
    OperationsList:= TList<TOperations>.Create; //загружаем с трилиста на
чертеж
    //получение типа документа
    if (kDoc.DocumentType = ksDocumentAssembly) then
    begin
      //если документ сборки, вызов функции считывания со сборки
      ComponentsList:= GetAssemblyComponents(ComponentsList);
      //создание основной формы
      FMainTable:= TFMain.Create(ComponentsList, 1);
      FMainTable.Caption:= 'ТБ1 (со сборки)';
    end
    //если документ чертеж
    else if (kDoc.DocumentType = ksDocumentDrawing) then
    begin
      //обработчик ошибки при считывании таблицы
      try
        GetDrawingComponents(ComponentsList);
      except on E: Exception do
        showkompasmessage('Ошибка при считывании с чертежа. Не найдена
таблица. ' + #10 + e.Message,1);
      end;
      //создание основной формы
      FMainTable:= TFMain.Create(ComponentsList, 0);
      //название формы
      FMainTable.Caption:= 'ТБ1 (с чертежа)';
    end;
    //показываем форму
    FMainTable.Show;
  end
  else
  begin
    //если документ не является чертежом или сборкой вывод сообщения
    ShowKompasMessage('Откройте сборку или чертеж.', 2);
  end;
end;

```

Рис.3. Процедура модуля CreateLineDimension

```

function GetAssemblyComponents(List: TList<TComponents>): TList<TComponents>;
var
  c, i, a, level: integer;
  topPart: IPart7;
  Details : OleVariant;
  assembly : IKompasDocument3D;
  dispatchDetail : IDispatch;
  propMng : IPropertyMng;
  part7 : IPart7;
  Components: TComponents;
  subassembliesPartsList : TList<TPart>;
  propKeeper : IPropertyKeeper;
begin
  //получение интерфейсов для ко
  if KDoc <> nil then assembly:= kDoc as IKompasDocument3D;
  //лист для промежуточных данных
  subassembliesPartsList := TList<TPart>.Create;
  //менеджер свойств документа
  propMng := newKompasAPI as IPropertyMng;
  if (assembly <> nil) then
  begin
    //поиск свойства TEl, для автомат. проставления галочек в программе
    FindProperty(propMng, assembly);
    //получение первой детали в дереве
    topPart := GetTopPart7(assembly);
    if (topPart <> nil) then
    begin
      //массив деталей
      Details := TopPart.PartsEx[0];
      if VarType(Details) = (varArray or varDispatch) then
      begin
        c := VarArrayHighBound(Details, 1) - VarArrayLowBound(Details, 1);
        Components.designation:= topPart.Name;
        Components.notation:= topPart.Marking;
        Components.Atype:= 'Сборка';
        Components.count:= c;
        Components.level:= 0;
        List.Add(Components);
        //считывание свойств деталей из массива
        for i := 0 to c do
        begin
          //уровень вложенности деталей (в данный момент функция не нужна)
          level:= 0;
          dispatchDetail := Details[i];
          if(dispatchDetail <> nil)then
          begin
            part7 := dispatchDetail as IPart7;
            if (part7 <> nil) then
            begin
              subassembliesPartsList.Clear;
              //рекурсивное считывание дерева, если деталь имеет итемы, то
              //вызываем еще раз и записываем в лист
              Recurs(part7, subassembliesPartsList, level+1);
              for a := 0 to subassembliesPartsList.Count - 1 do
              begin
                propKeeper := subassembliesPartsList.Items[a].Part as
                IPropertyKeeper;
                //получение свойств - наименование, обозначение и материал и
                запись
                Components:= GetPropertiesFound(propKeeper, propMng,
                assembly, a, subassembliesPartsList);
                if Components.designation <> '' then
                begin
                  //добавление в лист
                  List.Add(Components);
                end;
              end;
            end;
          end;
        end;
      end;
      SortList:= TList<TComponents>.Create;
      //объединение одинаковых позиций
      List:= CompleteTable(List);

      subassembliesPartsList.Free;
      ComponentsCount:= List.Count;
      result:= list;
    end;
  end;
end;
end;
end;

```

Рис.5. Данные, считанные со сборки

```

procedure TFmain.FormCreate(Sender: TObject);
var
  p, i, count: integer;
  nd: TcxTreeNode;
  s: string;
  col: TcxTreeListColumn;
begin
  SetColumnFormat; //включаем колонку с номерами
  TreeList:= TList<TNodeRec>.create;
  tlMaterials.BeginUpdate;
  tlMaterials.Clear;
  tlMaterials.DefaultRowHeight:= 30;

  if _TypeDoc = 1 then
  begin
    _ComponentsList:= SortingNotation(_ComponentsList);

    if stAssembly then
    begin
      nd := tlMaterials.Add;
      nd.Values[tlMaterialsType.ItemIndex]:= 'kompas';
      nd.Values[tlMaterialsDES.ItemIndex]:= _ComponentsList.Items[0].notation
+ ' ' + _ComponentsList.Items[0].designation;
      nd.Values[tlMaterialsType.ItemIndex]:= 'Сборка';

      count:= 0;
      for p := 1 to _ComponentsList.Count - 1 do
        if _ComponentsList.Items[p].level = 1 then
          count:= count + 1;

      i:=1;
      while i <> count+1 do
      begin
        Recurs(nd, i, _ComponentsList);
        i:= i + 1;
      end;
    end
  else
  begin
    for i := 0 to _ComponentsList.Count - 1 do
    begin
      if _ComponentsList.Items[i].atype <> 'Сборка' then
      begin
        nd := tlMaterials.Add;
        if (_ComponentsList.Items[i].bool = 'Нет') or
        (_ComponentsList.Items[i].bool = '') then
        begin
          nd.Values[tlMaterialsCHECK.ItemIndex]:= false;
        end
        else
        begin
          nd.Values[tlMaterialsCHECK.ItemIndex]:= true;
        end;
      end;
    end;
  end;
end;

```

Рис.6. Процедура FormCreate (начало кода)

```
nd.Values[tlMaterialsType.ItemIndex]:= 'kompas';
nd.Values[tlMaterialsDes.ItemIndex]:=
_ComponentsList.Items[i].designation;
nd.Values[tlMaterialsNOT.ItemIndex]:=
_ComponentsList.Items[i].notation;
nd.Values[tlMaterialsGOST.ItemIndex]:=
MaterialCut(_ComponentsList.Items[i].material, 'GOST');
nd.Values[tlMaterialsMAT.ItemIndex]:=
MaterialCut(_ComponentsList.Items[i].material, 'MAT');
nd.Values[tlMaterialsAMO.ItemIndex]:=
_ComponentsList.Items[i].amount;
end;
end;
end;
SetNumber;
end
else if _TypeDoc = 0 then
begin
AddOperations(OperList);
for i := 0 to _ComponentsList.Count - 1 do
begin
nd:= tlMaterials.Add;
if (_ComponentsList.Items[i].designation = '-----') and
(_ComponentsList.Items[i].notation = '-----') and
(_ComponentsList.Items[i].material = '-----') then
begin
nd.Values[tlMaterialsType.ItemIndex]:= '';
end
else
nd.Values[tlMaterialsType.ItemIndex]:= 'kompas'; //Чтобы можно
было удалить строки обозначающие перенос на некот лист
CheckStrikeOutCell(i, _ComponentsList);
nd.Values[tlMaterialsCHECK.ItemIndex]:= true;
nd.Values[tlMaterialsNUM.ItemIndex]:= _ComponentsList.Items[i].Number;
//nd.Values[tlMaterialsNUM.ItemIndex]:= inttostr(i+1);
nd.Values[tlMaterialsDes.ItemIndex]:=
_ComponentsList.Items[i].designation;
nd.Values[tlMaterialsNOT.ItemIndex]:=
_ComponentsList.Items[i].notation;
nd.Values[tlMaterialsAMO.ItemIndex]:= _ComponentsList.Items[i].amount;
nd.Values[tlMaterialsMAT.ItemIndex]:=
_ComponentsList.Items[i].material;
nd.Values[tlMaterialsGOST.ItemIndex]:= _ComponentsList.Items[i].GOST;
if stAddColumnNote then
nd.Values[tlMaterials.ColumnByName('tlMaterialsNOTE').ItemIndex]:=
_ComponentsList.Items[i].Note;
for p := 0 to
length(_ComponentsList.Items[i].Operations){Length(ArrColOperation)} - 1 do
begin
nd.Values[ArrColOperation[p].ItemIndex]:=
_ComponentsList.Items[i].Operations[p];
end;
end;
end;
tlMaterials.EndUpdate;
end;
```

Рис.7. Процедура FormCreate (конец кода)

ТБ1 (со сборки)

№п	Наименование детали или сборочной единицы	Количество деталей	Обозначение чертежа	Материал	
				Обозначение стандарта	Материал группа испытаний
<input type="checkbox"/>	Фланец	1	23.01.01.00.00	Полиамид П-125-20 ТУ 6-05-898-73	
<input type="checkbox"/>	Пропеллер	1	23.01.03.00.00	Полиамид П-125-20 ТУ 6-05-898-73	
<input type="checkbox"/>	Крышка-гнездо	1	23.01.04.00.00	Полиамид П-125-20 ТУ 6-05-898-73	
<input type="checkbox"/>	Сопло	1	23.01.05.00.00	ГОСТ 1050-2013	Сталь 10
<input type="checkbox"/>	Термопара	1	23.01.06.00.00	ГОСТ 860-75	Олово
<input type="checkbox"/>	Кожух	1	23.02.00.00.00	Полиамид П-125-20 ТУ 6-05-898-73	
<input type="checkbox"/>	Выключатель ВНК-2000	1	23.02.01.00.00	Полиамид П-125-20 ТУ 6-05-898-73	
<input type="checkbox"/>	Клавиша	1	23.02.02.00.00	Полиамид П-125-20 ТУ 6-05-898-73	
<input type="checkbox"/>	Трубка	1	23.03.00.00.00	Смесь резиновая 3063 ТУ 38-1051082-86	
<input type="checkbox"/>	Клениник	1	23.04.00.00.00	ГОСТ 16338-85	Полиэтилен НД
<input type="checkbox"/>	Корпус-основа	1	23.05.00.00.00	Полиамид П-125-20 ТУ 6-05-898-73	
<input type="checkbox"/>	Корпус-крышка	1	23.06.00.00.00	Полиамид П-125-20 ТУ 6-05-898-73	
<input type="checkbox"/>	Кольцо	1	23.07.00.00.00	ГОСТ 1050-2013	Сталь 10

Сформировать таблицу

Добавить операции

Добавить строку

Сравнить со сборкой

Рис. 8. Данные, считанные со сборки в интерфейсе приложения

Интерфейс окна приложения (Рис. 8) сдержит панель со следующими кнопками:

1) «Сформировать таблицу» - вызов процедуры, отвечающей за формирование таблицы на чертеже. После нажатия на эту кнопку запускается процедура CreateDrawingDocument, которая создает документ чертежа и выполняет несколько проверок (Рис.9, Рис.10). Процедура создания нового листа, если данные не помещаются на одном листе, показана на Рис. 11

```

procedure CreateDrawingDocument;
var
  i, rowCount, SheetNumb: Integer;
  KDoc: IKompasDocument;
  kViews: IViews;
  kView: IView;
  LSheet: ILayoutSheet;
  LSheets: ILayoutSheets;
  DManager: IViewsAndLayersManager;
  kStamp: IStamp;
  nd: tdxTreeListNode;
  sformat: ShortInt;
begin
  if newKompasAPI <> nil then
  begin
    KDoc:= newKompasAPI.ActiveDocument;
    if KDoc.DocumentType = ksDocumentDrawing then
    begin
      if KDoc <> nil then KompasDocument2d:= KDoc as IKompasDocument2D;
      if KompasDocument2d <> nil then DManager:=
KompasDocument2d.ViewsAndLayersManager;
      if DManager <> nil then kViews:= DManager.Views;
      LSheets:= KompasDocument2d.LayoutSheets;

      if CheckTableOnDrawing(kViews) then
      begin
        if newKompasAPI.MessageBoxEx('Ведомость уже сформирована, вы хотите
перезаписать данные?',
'Sообщение',MB_YESNO + MB_ICONINFORMATION ) = idYes then
        begin
          ViewsDelete(kViews, LSheets);
          setSheetFormat(LSheets.Item[0]);
        end
        else
        begin
          exit;
        end;
      end
      else
      begin
        ViewsDelete(kViews, LSheets);
      end;

      if LSheets.Count = 1 then
      begin
        setSheetFormat(LSheets.Item[0]);
      end;
    end
    else
    begin
      KDoc:= newKompasAPI.Documents.Add(ksDocumentDrawing, true);
      if KDoc <> nil then KompasDocument2d:= KDoc as IKompasDocument2D;
      LSheet:= KompasDocument2d.LayoutSheets.Item[0];
      if LSheet <> nil then
      begin
        setSheetFormat(LSheet);
        LSheet.Format.VerticalOrientation:= false;
        kStamp:= LSheet.Stamp;
        kStamp.Text[2].Str:= ComponentsList.Items[0].notation + ' ' +
ComponentsList.Items[0].designation;
        kStamp.Text[1].Str:= 'Таблица контроля качества основного металла';
        kStamp.update;
        LSheet.Update;
      end;
    end;
  end;
end;

```

Рис. 9. Процедура CreateDrawingDocument (начало кода)


```

if stAssembly then
begin
nd:= FMaintable.tlMaterials.Items[0];
TreeListrecurs(nd, OperationsList);
end
else
begin
TreeListGetValues(OperationsList);
DeleteDelimiter(OperationsList);
end;

SheetNumb:= 0;
while OperationsList.count <> 0 do //Создание и заполнение таблицы
begin
if SheetNumb = 0 then
begin
CreateTable(SheetNumb, stCountStrFirst);
rowCount:= DeleteRowTable(table.RowsCount, 64, SheetNumb+1);
end
else
begin
CreateNewList(KompasDocument2d); //Создание нового листа
CreateTable(SheetNumb, stCountStrNext); //создание вида и таблицы на
нем
rowCount:= DeleteRowTable(table.RowsCount, 22, SheetNumb+1);
end;
for i := 0 to rowCount - 5 do
begin
OperationsList.Delete(0);
end;
SheetNumb:= sheetnumb + 1;
end;
end;
OperationsList.clear;
end;

```

Рис.10. Процедура CreateDrawingDocument (конец кода)

```

Procedure CreateNewList(KDoc2D: IKompasDocument2D); //Создание нового листа
var
LSheet: ILayoutSheet;
begin
//ChDir('..\Sys\');
LSheet:= KDoc2D.LayoutSheets.Add;
if LSheet <> nil then
begin
if (FileExists(TPath.Combine(GetCurrentDir, 'GRAPHIC.LYT'), true)) then
LSheet.LayoutLibraryFileName:= 'C:\Program Files\ASCON\KOMPAS-3D
v22\Sys\graphic.lyt';
LSheet.LayoutStyleNumber:= 2;
if stSheetFormat = 'A3' then
LSheet.Format.Format:= ksFormatA3
else if stSheetFormat = 'A4x3' then
begin
LSheet.Format.Format:= ksFormatA4;
LSheet.Format.FormatMultiplicity:= 3;
end
else if stSheetFormat = 'A4x4' then
begin
LSheet.Format.Format:= ksFormatA4;
LSheet.Format.FormatMultiplicity:= 4;
end;
LSheet.Format.VerticalOrientation:= false;
LSheet.Update;
end;
end;
end;

```

Рис.11. Процедура создания нового листа

Процедура создания самой таблицы приведена на Рис. 12 - Рис. 14.

```

procedure CreateTable(SheetNumb, RowCount: integer); //Создание и заполнение
таблицы
var
del: integer;
begin
if RowCount > OperationsList.count then
RowCount:= OperationsList.Count;

del:= FindDelimiter;
if del > 0 then
begin
if del <= RowCount then
begin
RowCount:= del;
OperationsList.Delete(del);
end;
end;
end;

DrawTable(CreateView(KompasDocument2d, SheetNumb), RowCount);
end;

```

Рис.12. Процедура создания таблицы (начало кода)

```

function DrawTable(kView: IView; RowCount: integer): integer; //Рисование
таблицы и заполнение шапки

procedure CreateNumberColumn;
var
  i: integer;
begin
  table.AddColumn(0, false);
  AddCell(0,0,11,69,table,2);
  AddCell(0,1,29,69,table,2);
  AddText('Мн',0,0,3.5,table, false, false);
  for i := 0 to RowCount - 1 do
    AddText(inttostr(i+1),i+4,0,3.5,table, False, false);
  DrawingTable.Update;
end;

var
  i, n, ColCount: integer;
  kf: double;
  DrawSymbCont: ISymbols2DContainer;
  iCellBound: ICellBoundaries;
begin
  if kView <> nil then DrawSymbCont:= kView as ISymbols2DContainer;
  DrawingTable:= DrawSymbCont.DrawingTables.Add(4, 1, 8, 5, ksTTLFirstRow);
  DrawingTable.x:= 23;
  DrawingTable.y:= 276;

  table:= DrawingTable as ITable;
  for i := 0 to 4 do
    table.AddColumn(table.ColumnsCount - 1,true);

  table.range[0,3,0,4].CombineCells;
  table.range[0,0,3,0].CombineCells;
  table.range[0,1,3,1].CombineCells;
  table.range[0,2,3,2].CombineCells;
  table.range[1,3,3,3].CombineCells;
  table.range[1,4,3,4].CombineCells;

  OperCount:= Trunc((FMainTable.tlMaterials.Bands.Count - 8)/2);
  if stAddColumnNote then
  begin
    if stSheetFormat = 'A4x3' then
      kf:= Simpleroundto(386/Opercount, -2)
    else if stSheetFormat = 'A3' then
      kf:= Simpleroundto(176/Opercount, -2)
    else if stSheetFormat = 'A4x4' then
      kf:= Simpleroundto(596/Opercount, -2);
    ColCount:= FMainTable.tlMaterials.Bands.Count - 2;
  end
  else
  begin
    if stSheetFormat = 'A4x3' then
      kf:= Simpleroundto(416/Opercount, -2)
    else if stSheetFormat = 'A3' then
      kf:= Simpleroundto(206/Opercount, -2)
    else if stSheetFormat = 'A4x4' then
      kf:= Simpleroundto(626/Opercount, -2);
    ColCount:= FMainTable.tlMaterials.Bands.Count - 1;
  end;
  if stReadColumnWidth and (FMainTable.Caption = 'ТБ1 (с чертёжа)') then
  begin
    AddCell(0,0,WDesignation,69,table,2); // Наименование
    AddCell(0,1,WAmount,69,table,2); // Количество
    AddCell(0,2,WNotation,69,table,1); // Обозначение
    AddCell(1,3,WMaterial,69,table,2); // Материал
    AddCell(1,4,WGost,69,table,2); // ГОСТ
  end
  else
  begin
    //if stAddColumnNumber then AddCell(0,0,40,69,table,2); // Номер
    AddCell(0,0,40,69,table,2); // Наименование
    AddCell(0,1,11,69,table,2); // Количество
    AddCell(0,2,55,69,table,1); // Обозначение
    AddCell(1,3,44,69,table,2); // Материал
    AddCell(1,4,32,69,table,2); // ГОСТ
  end;

  AddCell(1,5,kf,47,table,0);
  AddCell(3,5,kf,7,table,2);
  AddCell(2,0,0,Table,2);

```

Рис.13. Процедура создания таблицы (продолжение кода)

```

AddText('Наименование детали или сборочной единицы',0,0,3.5,table, false,
false);
AddText('Количество деталей',0,1,3.5,table, true, false);
AddText('Обозначение чертежа',0,2,5,table, false, false);
AddText('Материал',0,3,5,table, false, false);
AddText('Материал, группа испытаний',1,3,3.5,table, false, false);
AddText('Обозначение стандарта',1,4,3.5,table, false, false);
AddText('Наименование операции',0,5,5,table, false, false);
AddText('Шифр операции',2,5,3.5,table, false, false);

for i := 0 to OperCount - 2 do //колонки для операций
    table.AddColumn(table.ColumnsCount - 1,true);
    table.range[0,5,0,table.ColumnsCount - 1].CombineCells;
    table.range[2,5,2,table.ColumnsCount - 1].CombineCells;

for i := 8 to Round(OperCount) + 7 do //название операции
begin
    AddText(FMainTable.tlMaterials.Bands[i].Caption.Text,1,i-3,3.5,table,
true, false);
end;

for i := Round(OperCount) + 8 to ColCount do //номер операции
begin
    AddText(FMainTable.tlMaterials.Bands[i].Caption.Text,3,i-3-
OperCount,3.5,table, false, false);
end;

if stAddColumnNumber then //если включена колонка номеров, то добавляем
слева
begin
    ColNum:= 1;
    table.AddColumn(0, false);
    if stReadColumnWidth and (FMainTable.Caption = 'ТБ1 (с чертежа)') then
    begin
        AddCell(0,0,WNumber,69,table,2);
        AddCell(0,1,WDesignation,69,table,2)
    end
    else
    begin
        AddCell(0,0,11,69,table,2);
        AddCell(0,1,29,69,table,2);
    end;
    AddText('№н',0,0,3.5,table, false, false);
end
else
begin
    ColNum:= 0;
end;

iCellBound:= Table.Cell[2,5+ColNum] as ICellBoundaries;
iCellBound.LineStyle[ksCBAllBorders]:= ksCSNormal;

if stAddColumnNote then
begin
    table.AddColumn(table.ColumnsCount - 1, true);
    table.range[0,table.ColumnsCount - 1,3,table.ColumnsCount -
1].CombineCells;
    if stReadColumnWidth then
        AddCell(0,table.ColumnsCount - 1,WNote,69,table,2)
    else
        AddCell(0,table.ColumnsCount - 1,30,69,table,2);
    AddText('Примечание',0,table.ColumnsCount - 1,3.5,table, false, false);
end;

if stReadColumnWidth and (Length(WArray) <> 0) and (FMainTable.Caption =
'TБ1 (с чертежа)') then //считывание длины колонок
    SetOperationWidth;

for i := 3 to RowCount - 1 + 3 do
begin
    table.AddRow(i,true);
end;
DrawingTable.Update;

AddToTable(OperationsList, RowCount); //Добавление текста в таблицу
DrawingTable.Update;

end;

```

Рис.14. Процедура создания таблицы (конец кода)

2) «Добавить операции» - добавление операций и их шифров. При нажатии на данную кнопку появляется новая форма «Выбор операций» (Рис. 15), где пользователь выбирает необходимые операции или добавляет новые. При нажатии на «Добавить операцию» открывается окно с формой для заполнения, в которую пользователь вводит наименование операции и ее шифр. Отмена добавления операции осуществляется с помощью кнопки «Отмена» или закрытием активной формы. Нажатие кнопки «Ок» запускает процедуру добавления операции в форму, операция добавляется в конец списка и выводится в отсортированном виде по шифру.

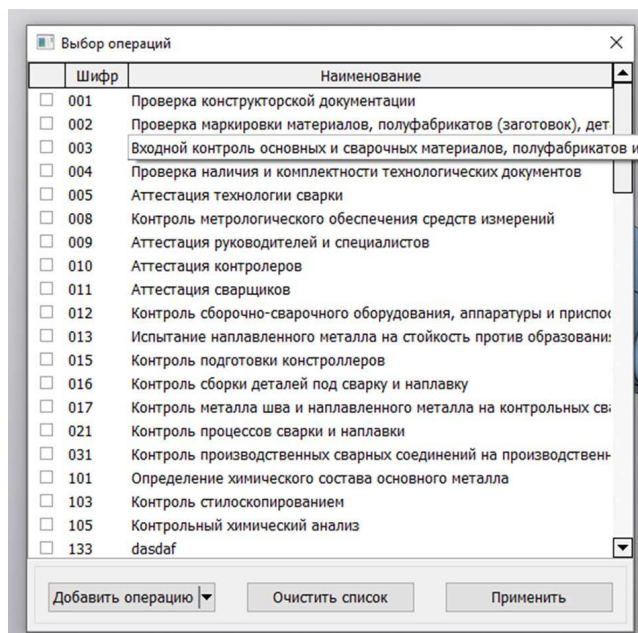


Рис.15. Форма «Выбор операций»

Нажатие на кнопки «Применить» приводит к тому, что форма «Выбор операций» закрывается и операции добавляются в основную форму (Рис. 16).

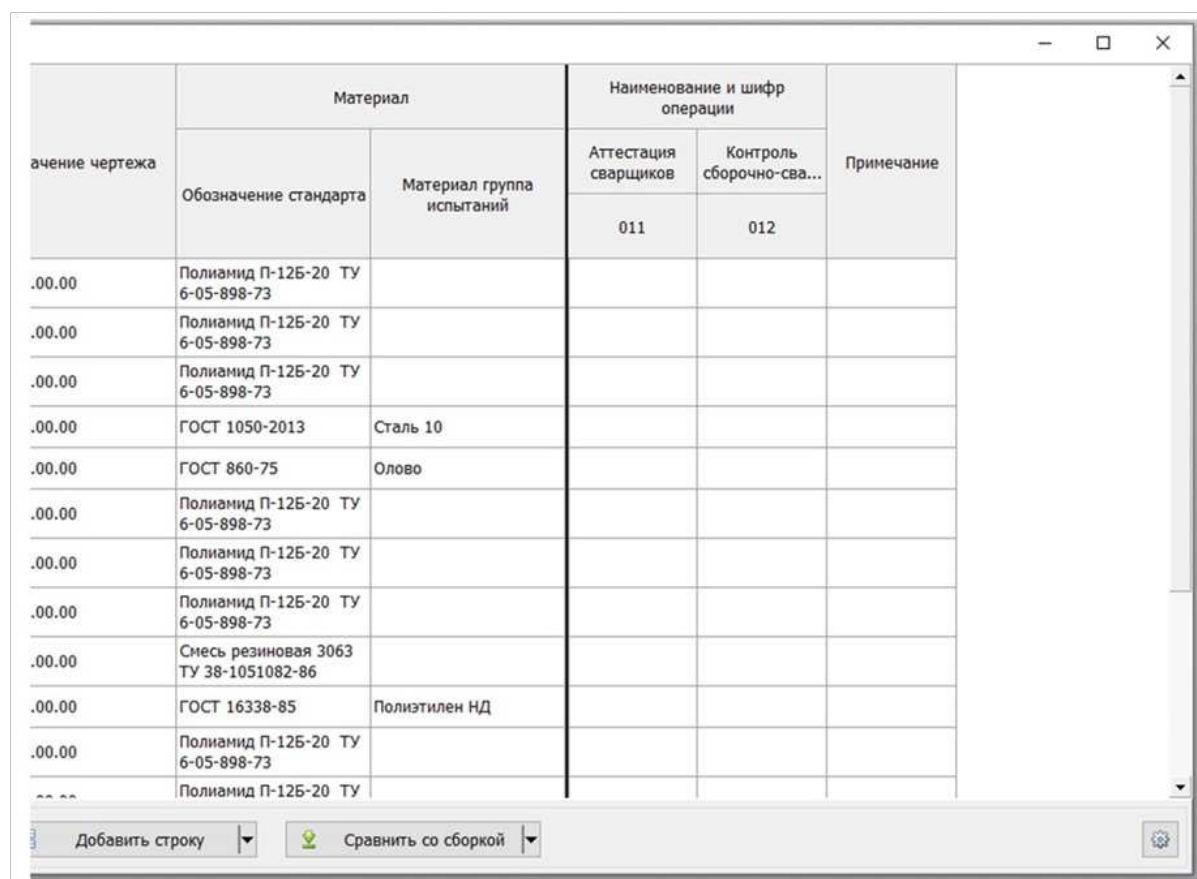


Рис.16. Добавленные операции в интерфейсе приложения

Нажатие на ячейку под операцией, приводит к появлению выпадающего списка с символами для вставки в эту ячейку.

Все операции хранятся в текстовом файле, который находится в каталоге с приложением (Рис. 17). При запуске, приложение считывает все операции и заносит в форму, после закрытия операции сохраняются в файле OperationList.txt.

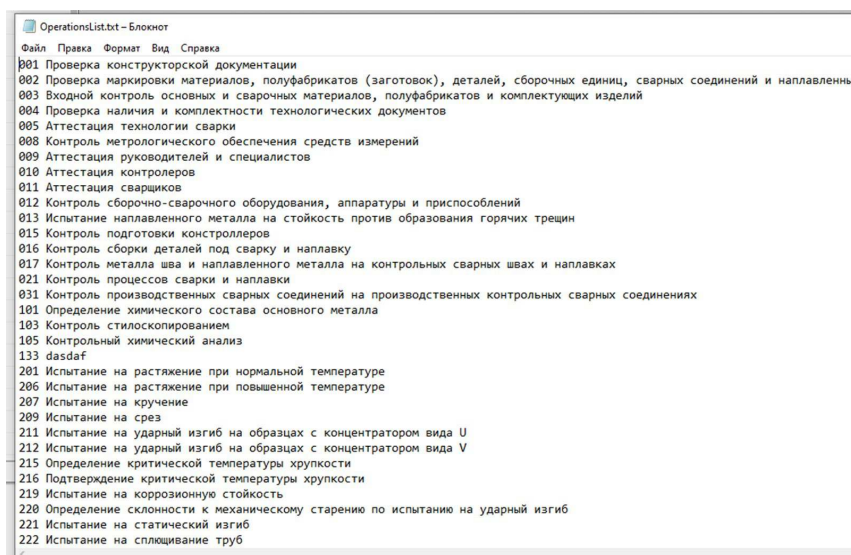


Рис.17. Файл с операциями

Процедуры, которые срабатывают после нажатия на кнопку «Применить», показаны на Рис. 18, Рис. 19.

```
procedure TFOperations.bApplyClick(Sender: TObject);
var
  i: Integer;
  operRec: TOper;
begin
  for i := 0 to tloper.Count - 1 do
  begin
    //проверка на выбранный компонент
    if not varIsNull(tloper.Items[i].Values[tlColChecks.ItemIndex]) then
      if tloper.Items[i].Values[tlColChecks.ItemIndex] then
      begin
        //запись в лист
        operRec.name:=
        varToStr(tloper.Items[i].Values[tlColOperation.ItemIndex]);
        operRec.number:=
        varToStr(tloper.Items[i].Values[tlColNumber.ItemIndex]);
        operRec.bool:= true;
        operlist.Add(operrec);
      end;
    end;
  end;
  //процедура добавления операций на главной форме
  FmainTable.AddOperatio(OperList);
  //закрытие формы операций
  self.Close;
end
```

Рис.18. Процедуры, срабатывающие при нажатии на кнопку «Применить» (начало кода)

```

procedure TFmain.AddOperations(list: TList<TOper>);
var
  p, i, ln: integer;
  AColumn, LastCol: TcxTreeListColumn;
  AComboProperties: TcxComboBoxProperties;
  CF: TCharFormat;
  MS: TMemoryStream;
begin
  if tlMaterials.Bands.Count > 7 then //удаление бендов операций
  begin
    for i := 7 to tlMaterials.Bands.Count - 1 do
    begin
      tlMaterials.Bands.Items[7].Destroy;
    end;

    for i := 0 to length(ArrColOperation) - 1 do
    begin
      ArrColOperation[i].Destroy;
    end;

    SetLength(ArrColOperation, 0);
    try
      tlMaterials.ColumnByName('tlMaterialsNOTE').Destroy;
    except
    end;
  end;

  //Если лист пустой завершаем процедуру
  if list.Count = 0 then
    exit;

  BandCreate('Наименование и шифр операции', -1, 0, false);
  for i := 0 to list.Count - 1 do //бенды названий операций
  begin
    BandCreate(list.Items[i].name, 7, i+1, true);
  end;

  SetLength(ArrColOperation, list.Count);
  for i := 0 to list.Count - 1 do //бенды с шифрами операций
  begin
    BandCreate(list.Items[i].number, 8+i, 0, false);
  end;

  AColumn:= tlMaterials.CreateColumn();
  ArrColOperation[i]:= AColumn;

  //настройки колонки: позиция, шрифт, класс и тд
  AColumn.Position.BandIndex:= 8+list.count+i;
  AColumn.DataBinding.ValueTypeClass := TcxStringValue;
  AColumn.PropertiesClass := TcxComboBoxProperties;
  AComboProperties := AColumn.Properties as TcxComboBoxProperties;
  AComboProperties.Items.Add('+C');
  AComboProperties.Items.Add('-');
  AComboProperties.Items.Add('+$cn./it.7$');
  AColumn.Options.VertSizing:= false;
  AColumn.Properties.Alignment.Horz:= taCenter;
end;
//если включена настройка "колонка примечание"
if stAddColumnNote then //бенд "примечание"
begin
  BandCreate('Примечание', -1, 6, false);
  AColumn:= tlMaterials.CreateColumn();
  AColumn.Position.BandIndex:= tlMaterials.Bands.Count - 1;
  AColumn.name:= 'tlMaterialsNOTE';
end;
end;

```

Рис.19. Процедуры, срабатывающие при нажатии на кнопку «Применить» (конец кода)

3) «Сравнить со сборкой» - сравнение считанных данных с теми, которые на текущий момент находятся в сборке, отличающиеся данные выделяются голубым цветом.

4) «Добавить строку» - добавление новой строки выше выбранной. Эта кнопка имеет выпадающий список, где находятся дополнительные кнопки для редактирования данных («Добавить строку снизу», «Начать с нового листа», «Удалить строку», «Выбрать все элементы», «Зачеркнуть текст в строке», «Зачеркнуть текст в ячейке»).

Результаты разработки. В результате последовательного выполнения шагов описываемого порядка разработки приложения, пользователь получит новый документ чертежа, со сформированной таблицей контроля качества основного металла ТБ1 (Рис. 20). Для примера, представленного на Рис.2, в таблице приводятся следующие сведения и данные: наименование детали или сборочной единицы, количество деталей, обозначение чертежа, материал (материал, группа испытаний; обозначение стандарта), наименование операции (входной контроль основных и сборочных материалов, полуфабрикатов и комплектующих изделий; проверка наличия и комплектности технологических документов; контроль метрологического обеспечения средств измерений; аттестация руководителей и специалистов; аттестация контролеров; контроль сборочно-сварочного оборудования, аппаратуры и приспособлений; испытание

наплавленного металла на стойкость против образования горячих трещин).

		Материал		Наименование операции								
№п	Наименование детали или сборочной единицы	Количество деталей	Обозначение чертежа	Материал, группа испытаний	Обозначение стандарта	Визуальный контроль внешних и скрытых поверхностей поверхностей и контуров изделия	Проверка качества и количества технологических документов	Контроль температурного обеспечения средств измерений	Атмосферная влажность и спецификаций	Атмосферная влажность	Контроль сборочно-сварочных работ, операций и приспособлений	Испытание материала металла на прочность, прогиб, обломки, зазоры, трещины
						003	004	008	009	010	012	013
1	Фланец	1	23.0101.00.00		Полный П-125-20 ТУ 6-05-898-73							
2	Пропеллер	1	23.0103.00.00		Полный П-125-20 ТУ 6-05-898-73							
3	Крышка-гнездо	1	23.0104.00.00		Полный П-125-20 ТУ 6-05-898-73							
4	Сопло	1	23.0105.00.00	Сталь 10	ГОСТ 1050-2013							
5	Термопара	1	23.0106.00.00	Олово	ГОСТ 860-75							
6	Капучин	1	23.02.00.00.00		Полный П-125-20 ТУ 6-05-898-73							
7	Выключатель ВНС-2000	1	23.02.01.00.00		Полный П-125-20 ТУ 6-05-898-73							
8	Клавиша	1	23.02.02.00.00		Полный П-125-20 ТУ 6-05-898-73							

				23.00.00.00.00 Термистолет			
Лист	Лист	№ докум.	Подп.	Дата	Лист	Масса	Максимум
Рисунки							11
Проц.					Лист	Листов	1
Технот.							
Нормат.							
Экз.							

Таблица контроля качества основного металла

Рис. 20. Пример таблицы контроля качества основного металла

Разработанное прикладное пользовательское приложение автоматически считывает данные с файла сборки открытого в КОМПАС-3D, создает новый документ чертежа, осуществляет построение ТБ1, с занесением данных, взятых со сборочного документа.

Научная новизна полученных результатов. Разработанная процедура построения таблицы ТБ1 в КОМПАС-3D (в условиях ограниченной информации об алгоритмах составления таблиц контроля качества в свободном доступе и затруднениях возникающих у специалистов проектов, составляющих конструкторскую документацию таблицы ТБ1 и, на их основании, программ качества), в отличие от известных, позволяет осуществлять построение автоматически.

Заключение. В работе приведен порядок разработки прикладного пользовательского приложения для КОМПАС-3D, позволяющего в автоматизированном режиме сформировать таблицу контроля качества основного металла (ТБ1). Для реализации используется программный код, написанный в программной среде Delphi. Данное приложение может быть использовано предприятиями (организациями), специализирующимися на проектировании и изготовлении оборудования для атомной энергетики, АЭС.

Разработанное пользовательское приложение отвечает критерию гибкости, а также быстроты, в условиях возможности внесения оперативных изменений. Процесс разработки осуществлялся итеративно, что позволяло расширять функционал. Например, в качестве расширения возможностей кода, может быть предусмотрено использование процедуры считывания данных с одного листа (активным листом считается лист, выбранный в дереве) – приложение переходит в режим «ТБ1 Активный лист», в котором кнопка «Построить таблицу» заменяется на кнопку «Обновить данные на активном

листе». С помощью данного режима пользователь может легко изменить данные всего лишь на одном листе.

За счет универсальности программного кода, возможно его применение не только для КОМПАС-3D, но и для других CAD-систем.

Также в качестве перспектив дальнейшего использования для проектов по созданию прикладных библиотек для КОМПАС-3D могут быть взяты за основу принципы написания программного кода процедур разработки, что входит в область интересов как предприятий (например, для действующих конструкторов, специалистов, повышающих квалификацию в соответствующей области) и конструкторских бюро, так и учебных заведений.

Библиография

1. Норсеев С. Разработка приложений под КОМПАС в Delphi, 2013. – 346 с.[Электронный ресурс] – URL: <https://norseev.ru/books/kompas-delphi-2013/>
2. Сошкина Д.Е. Разработка пользовательской библиотеки в Компас-3D / Д.Е. Сошкина; науч. рук. А.А. Дронов // Прогрессивные технологии и экономика в машиностроении : сборник трудов XIII Всероссийской научно-практической конференции для студентов и учащейся молодежи, 7-9 апреля 2022 г., Юрга. – Томск: Изд-во ТПУ, 2022. – С. 34-36.
3. Гуца А.О. К задаче о разработке на С# пользовательских библиотек стандартных деталей для КОМПАС-3D / А.О. Гуца, Т.Д. Стасенко ; науч. рук. В.С. Мурашко // Исследования и разработки в области машиностроения, энергетики и управления : материалы XXIII Междунар. науч. – техн. конф. студентов, аспирантов и молодых ученых, Гомель, 27–28 апр. 2023 г. : в 2 ч. Ч. 2 / М-во образования Респ. Беларусь, Гомел. гос. техн. ун-т им. П.О. Сухого ; под общ. ред. А.А. Бойко. – Гомель: ГГТУ им. П. О. Сухого, 2023. – С. 268-270.
4. Терехин Р.Д., Фоломкин А.И. Разработка программных библиотек для демонстрации и построения линий пересечения различных поверхностей / Технические науки – от теории к практике // Сб. ст. по материалам LXXI междунар. науч. – практ. конф. № 6 (66). Новосибирск: Изд. АНС «СиБАК», 2017. – С. 13-18.
5. Решетникова Е.С., Савельева И.А., Свистунова Е.А. Геометрическое моделирование и разработка пользовательских библиотек при проектировании объектов машиностроения // Программные системы и вычислительные методы. 2020. № 1. С. 1-7. DOI: 10.7256/2454-0714.2020.1.32292 URL: https://nbpublish.com/library_read_article.php?id=32292
6. Кидрук М.И. Конструкторские библиотеки и инструменты для их создания в системе КОМПАС-3D. Часть 1. В помощь конструктору – библиотеки, расчетные модули и справочники КОМПАС-3D // статья в журнале «САПР и графика». Машиностроение, № 1, 2006 [Электронный ресурс] – URL: <https://sapr.ru/article/14850>
7. Кидрук М.И. Конструкторские библиотеки и инструменты для их создания в системе КОМПАС-3D. Часть 2. Сделай сам, или Как создать свою библиотеку для КОМПАС-3D // статья в журнале «САПР и графика». Машиностроение, № 2, 2006 [Электронный ресурс] – URL: <https://sapr.ru/article/15598>
8. Форум пользователей ПО Аскон. [Электронный ресурс] – URL: <http://forum.ascon.ru/index.php>
9. ОСТ 108.004.10-86. Отраслевой стандарт. Программа контроля качества изделий атомной энергетики. – Действует с 01.01.2007 – М.: Изменение № 12, 2019. 143 с. 1. Норсеев С. Разработка приложений под КОМПАС в Delphi, 2013. – 346 с.[Электронный ресурс] – URL: <https://norseev.ru/books/kompas-delphi-2013/>
10. РД ЭО 1.1.2.01.0713-2019. Оценка соответствия в формах приемки, испытаний

продукции для атомных станций. Положение. – Взамен РД ЭО 1.1.2.01.0713-2013 «Положение об оценке соответствия в форме приемки и испытаний продукции для атомных станций». – Введ. с 20.02.2020 – М.: Актуализация 01.01.2021. – 60 с.

11. Форум DWG.RU для обсуждения проблем проектирования, конструирования в области строительства, архитектуры, машиностроения, смежных отраслей. [Электронный ресурс] – URL: <https://forum.dwg.ru/showthread.php?t=167100>

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Представленная статья на тему «Разработка прикладного приложения для автоматизации проектирования в КОМПАС-3D: таблица контроля качества основного металла» соответствует тематике журнала «Программные системы и вычислительные методы» и посвящена вопросу разработки прикладного пользовательского приложения для Компас-3D позволяющее автоматически сформировать таблицы контроля качества основного металла (ТБ1). Авторами в статье указаны основные требования к функционалу пользовательского приложения, среди которых: запуск происходит со сборки в КОМПАС-3D через панель команд или через кнопки «Приложения» - «ТБ1»; после чего в окне программы отображается таблица с полученными со сборки деталями (также детали всех подборок), их количеством, обозначением, материалом и соответствующим стандартом; также программа должна быть функциональной, позволяющей конструктору менять все графы таблицы, выбирать те позиции, которые построятся в таблице на чертеже, добавлять новые позиции, удалять добавленные, зачеркивать текст и возможность разделить таблицу на несколько листов; в приложении должен быть предусмотрен выбор контрольных операций для материалов сборки, операций с шифрами, также добавление новых операций, с последующим их сохранением в памяти программы; иметься возможность выбора символов для операций по клику на соответствующую ячейку; по нажатию на соответствующую кнопку автоматически должен создаваться документ чертежа, с построенной таблицей со всеми изменениями, внесенными конструктором в интерфейс программы; в наличии должна быть возможность добавлять функционал считывания данных с чертежа для внесения изменений через интерфейс программы.

В статье авторами приведен порядок разработки прикладного пользовательского приложения для КОМПАС-3D, позволяющего в автоматизированном режиме сформировать таблицу контроля качества основного металла (ТБ1). Для реализации этого использовался программный код, написанный в программной среде Delphi, которая используется предприятиями (организациями), специализирующимися на проектировании и изготовлении оборудования для атомной энергетики, АЭС.

Стиль и язык изложения материала является достаточно доступным для широкого круга читателей. Практическая значимость статьи четко обоснована. Статья достаточно структурирована - в наличии введение, заключение, внутреннее членение основной части.

К недостаткам можно отнести следующие моменты: из содержания статьи не прослеживается научная новизна. Отсутствует четкое выделение предмета, объекта исследования. Статья не соответствует по объему минимальным требованиям (объем должен быть не менее 12 000 знаков).

Рекомендуется четко обозначить научную новизну исследования, сформулировать предмет, объект. Увеличить объем статьи до 12 000 знаков или более. Также будет

целесообразным добавить о перспективах дальнейшего исследования.

Статья «Разработка прикладного приложения для автоматизации проектирования в КОМПАС-3D: таблица контроля качества основного металла» требует доработки по указанным выше замечаниям. После внесения поправок рекомендуется к повторному рассмотрению редакцией рецензируемого научного журнала.

Результаты процедуры повторного рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Статья посвящена разработке прикладного пользовательского приложения для автоматизации составления таблицы контроля качества основного металла (ТБ1) в САПР КОМПАС-3D. В рамках работы рассматривается создание программного продукта, который ускоряет и упрощает процесс формирования документации в области проектирования оборудования для атомной энергетики.

Автор применяет метод разработки программного обеспечения с использованием языка Delphi. Статья подробно описывает последовательность действий по созданию и интеграции библиотеки для КОМПАС-3D, взаимодействие с его API, а также механизмы автоматизации работы с данными сборки. Программная часть включает описание ключевых процедур, таких как чтение данных из сборки, создание чертежей, а также функционал для редактирования таблиц.

Процесс проектирования и составления таблиц контроля качества в атомной энергетике является крайне важным, так как от точности и правильности документации зависит качество изготовления и безопасности эксплуатации оборудования. Разработка такого прикладного приложения отвечает актуальным потребностям отрасли, а также способствует ускорению и автоматизации сложных инженерных процессов. Это решение актуально в условиях современных требований к проектированию и документированию в атомной энергетике.

Научная новизна работы заключается в предложении автоматизированного подхода к формированию таблиц контроля качества в КОМПАС-3D, что значительно улучшает точность и снижает время на составление документации. В отличие от существующих решений, это приложение позволяет динамично изменять данные в процессе разработки и легко интегрировать новые функциональные возможности в приложение.

Статья написана четким и последовательным стилем, доступным для целевой аудитории, которая включает как разработчиков программного обеспечения, так и специалистов, занимающихся проектированием в КОМПАС-3D. Работа логично структурирована: введение, постановка задачи, описание методов разработки, реализация и результаты. Статья сопровождается подробными иллюстрациями, которые помогают визуализировать ключевые этапы разработки. Это делает материал не только теоретически значимым, но и практически полезным.

Автор тщательно подобрал литературу, включая как фундаментальные работы, так и современные исследования, связанные с разработкой приложений для САПР КОМПАС-3D. Библиография отражает разнообразие источников, что повышает научную ценность работы. Однако для повышения уровня работы рекомендовано добавить более актуальные источники, связанные с новыми технологиями в области автоматизации проектирования и разработки программного обеспечения.

Статья не содержит явных слабых мест. Применяемая методология и разработка показывают высокий уровень знаний автора в области программирования и проектирования в КОМПАС-3D. Возможные улучшения могут касаться детализации

алгоритмов обработки данных и их оптимизации, а также расширения функционала приложения для использования в более широких контекстах, помимо атомной энергетики.

Статья представляет собой ценный вклад в развитие программных средств для автоматизации инженерных процессов в атомной энергетике. Практическая значимость работы высока, так как разработанный инструмент позволяет повысить эффективность работы инженеров, а также снизить вероятность ошибок при составлении таблиц контроля качества. Работу можно рекомендовать как для специалистов в области проектирования, так и для ученых, занимающихся развитием программных систем для инженерных задач. Статья может быть интересна широкой аудитории, включая как профессионалов, так и студентов, изучающих программное обеспечение для САПР.

Работа заслуживает публикации с учетом ее научной значимости и практической полезности. Рекомендуется принять статью к публикации с незначительными доработками, связанными с улучшением анализа существующих решений в области автоматизации проектирования и расширением функционала описанного приложения.

Программные системы и вычислительные методы

Правильная ссылка на статью:

Плетнева А.О., Янчус В.Э. Разработка методики тестирования мобильного приложения на основе технологии ай-трекинга // Программные системы и вычислительные методы. 2025. № 1. DOI: 10.7256/2454-0714.2025.1.69889 EDN: VCNGRM URL: https://nbpublish.com/library_read_article.php?id=69889

Разработка методики тестирования мобильного приложения на основе технологии ай-трекинга

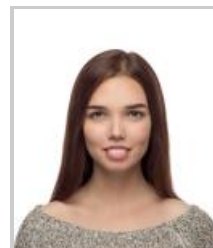
Плетнева Алиса Олеговна

ORCID: 0009-0009-1502-2660

магистр, Высшая школа дизайна и архитектуры, Санкт-Петербургский политехнический университет
Петра Великого

194064, Россия, г. Санкт-Петербург, ул. Политехническая, 29, оф. 29АФ

✉ pletneva.alisa2010@yandex.ru



Янчус Виктор Эдмундасович

ORCID: 0000-0001-7220-0819

кандидат технических наук

преподаватель Высшей школы дизайна и архитектуры Санкт-Петербургского политехнического
университета Петра Великого

194064, Россия, г. Санкт-Петербург, ул. Политехническая, 29, оф. 29АФ

✉ victorimop@mail.ru



[Статья из рубрики "Компьютерная графика, обработка изображений и распознавание образов"](#)

DOI:

10.7256/2454-0714.2025.1.69889

EDN:

VCNGRM

Дата направления статьи в редакцию:

18-02-2024

Аннотация: Предметом исследования выступают средства компьютерной графики для разработки мобильного приложения. Объектом исследования является мобильное приложение. Подробно с точки зрения дизайна рассматриваются 4 фактора влияния на восприятие информации в мобильном приложении: цветовое решение, композиционное решение, шрифт и количество букв в управляющих словах. Был разработан

эксперимент, который проводился с помощью технологии ай-трекинга в специальном программном модуле. Он включает в себя 72 стимула и 1 поставленную задачу, которую необходимо решить участникам эксперимента. Особое внимание уделяется особенности мобильного приложения, она заключается в том, что вся информация находится в поле центрального зрения человека, периферийное зрение при работе в этой среде слабо задействуется. Следовательно, влияние саккад на шаблон рассматривания интерфейса мобильного приложения значительно меньше, чем при взаимодействии человека с графическим интерфейсом приложений и сайтов, ориентированных на воспроизведение на большем экране персонального компьютера. Технология ай-трекинга и методы математической статистики для проведения экспериментальных исследований восприятия визуальной информации человеком в графическом интерфейсе мобильных приложений. Научная новизна заключается в применении технологии ай-трекинга для экспериментального исследования восприятия визуальной информации человеком в графическом интерфейсе мобильных приложений. Быстрее всего участники эксперимента решали задачу в сине-оранжевом и бело-черном цветовых сочетаниях стимульного материала, а также наиболее быстро испытуемые решали задачи с крупными активными элементами в композиционном решении. Также проявляется статистически значимое влияние шрифта с засечками. При его использовании в стимульном материале испытуемые решают задачу эксперимента быстрее. Интересным наблюдением стало то, что длинные слова воспринимались испытуемыми быстрее коротких. Таким образом, все выделенные факторы: цветовое решение, композиционное решение, шрифт и количество букв в управляющих словах влияют на восприятие в мобильном приложении, но при определенных условиях. Результаты работы: 1. Разработанная методика продемонстрировала свою работоспособность; 2. Данная методика может быть использована для тестирования мобильных приложений; 3. Данная методика позволяет разработать мобильное приложение с учетом выделенных рекомендаций; 4. Необходимо разрабатывать следующие задачи для проведения новых экспериментов.

Ключевые слова:

Мобильное приложение, Технология ай-трекинга, Фиксации, Саккады, Центральное зрение, Периферийное зрение, Цветовое решение, Композиционное решение, Шрифт, Управляющие слова

В настоящее время мы регулярно взаимодействуем с различными гаджетами: телевизор, компьютер, планшет, мобильный телефон. Последний используется ежедневно для различных задач, например, для мобильной связи, для общения в мессенджерах и социальных сетях, а также для мобильных приложений [\[1\]](#). Кроме того, выход в Интернет является востребованной функцией современного смартфона, так как с его помощью можно в короткие сроки получить доступ к необходимой информации, развлекательному контенту, и приобрести необходимые товары и услуги [\[2\]](#). Смартфон – довольно небольшое устройство, вмещающее в себя много информации, которую необходимо обрабатывать и анализировать для достижения пользователем поставленной цели. По этой причине многие компании разрабатывают мобильные приложения [\[3,4\]](#). Мобильное приложение – это специальная программа для смартфона, установленная на определенной платформе (наиболее распространенные платформы: IOS и Android), обладающая функционалом, позволяющим выполнять определенные действия в

зависимости от тех задач, которые необходимо решить [\[2,5\]](#).

При проектировании мобильных приложений необходимо использовать узнаваемые пользователю элементы. Для этого мобильное приложение должно обладать следующими характеристиками:

1. Очевидность и ясность для пользователя;
- 2 . Понятность. Пользователи должны понимать, с какими элементами они взаимодействуют;
- 3 . Предсказуемость в процессе взаимодействия с элементами мобильного приложения [\[6\]](#).

Мобильное приложение нужно разрабатывать максимально простым и понятным, чтобы у пользователя сохранялось ощущение прямого управления объектом его внимания [\[7\]](#).

При проектировании мобильного приложения основными элементами, с точки зрения дизайна, являются: цветовое решение, композиционное решение и выбор шрифта [\[8\]](#). Также мобильное приложение наполняется текстом. Текст должен быть четким, понятным и структурированным. Пользователь может находиться в различных условиях наблюдения: смотреть в экран телефона под ярким солнцем на улице, на бегу или с низкой яркостью экрана по причине низкого заряда. Все эти факторы могут сильно влиять на восприятие текста и прочей информации [\[9\]](#).

Цель работы – разработка методики исследования восприятия визуальной информации в мобильном приложении. Для достижения цели были поставлены следующие задачи:

- разработать стимульный материал;
- разработать задачу для испытуемых;
- создать эксперимент в специализированном программном модуле;
- проанализировать результаты, полученные в ходе проведенного эксперимента, средствами математической статистики.

В качестве гипотезы выдвигается предположение о том, что все факторы: цветовое решение, композиционное решение, шрифт и количество букв в управляющих словах влияют на скорость решения поставленной задачи испытуемым при взаимодействии с графическим интерфейсом мобильного приложения.

В настоящей работе будет рассматриваться влияние факторов цветового решения, композиционного решения, типа шрифта и количества букв в управляющих словах на скорость решения условной задачи испытуемым при рассматривании стимула, имитирующего интерфейс мобильного приложения, при помощи технологии ай-трекинга (Рис. 1).

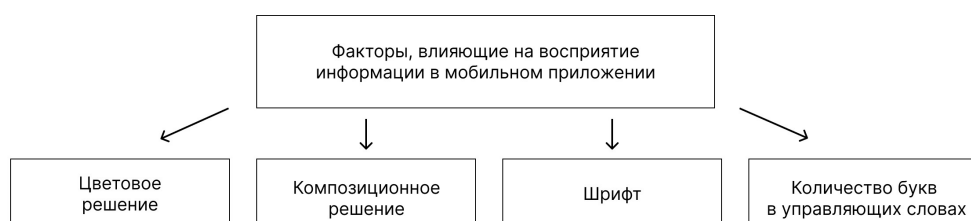


Рис. 1 – Факторы, влияющие на восприятие информации в мобильном приложении

Технология ай-трекинга используется в различных областях жизнедеятельности человека: в образовании, в науке, в маркетинге. В последнем данная технология используется для тестирования рекламы, упаковки, юзабилити сайтов при помощи стационарного или мобильного оборудования – ай-трекера [\[10,11\]](#).

Среди параметров шаблона рассматривания наибольший интерес вызывают фиксации – области, в которых взгляд пользователя задерживается на продолжительное время, в течение которого происходит восприятие информации. Процесс движения взгляда пользователя между точками фиксации называется «саккада». Воспроизведение саккад показывает, какой путь проделывает взгляд между точками фиксации [\[12\]](#).

Особенность мобильного приложения заключается в том, что вся информация находится в поле центрального зрения человека, периферийное зрение при работе в этой среде слабо задействуется (Рис. 2) [\[13,14\]](#). Следовательно, влияние саккад на шаблон рассматривания интерфейса мобильного приложения значительно меньше, чем при взаимодействии человека с графическим интерфейсом приложений и сайтов, ориентированных на воспроизведение на большем экране персонального компьютера.

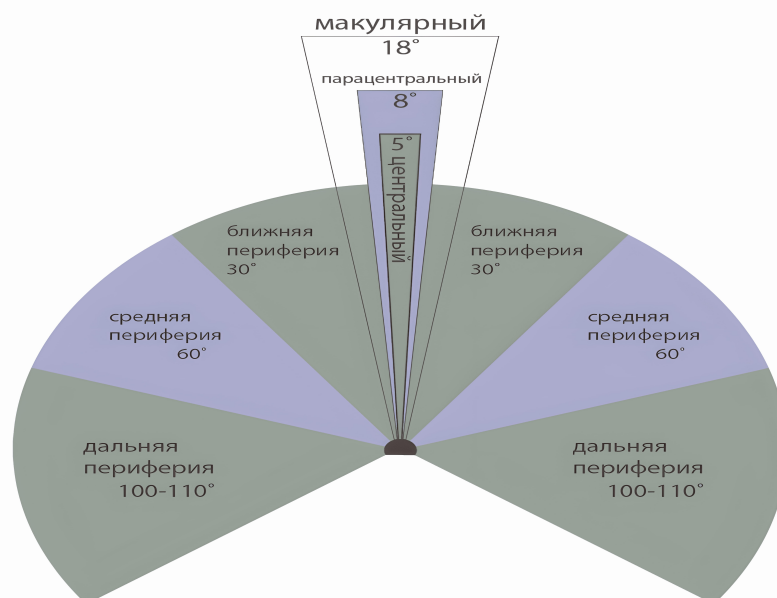


Рис. 2 – Области зрения человека

В настоящей работе в качестве экспериментальной установки был применен программно-аппаратный комплекс ай-трекинга. С помощью данного оборудования создается набор данных, записывая координаты определенное количество раз в секунду [\[15\]](#). Затем этот набор данных можно визуализировать и интерпретировать, чтобы выявить поведение, незаметное для пользователя. Отслеживание взгляда (ай-трекинг, окулография) – это технология, которая позволяет узнать направление взгляда человека, предмет его интереса и продолжительность фокусировки [\[16,17,18\]](#). Прибор ай-трекер позволяет получить набор параметрических данных в виде координат движения взора человека по стимульному материалу [\[19\]](#). Далее эти экспериментальные данные подвергаются статистическому анализу с целью выявления статистически значимых зависимостей влияния факторов на параметры шаблона рассматривания стимульного материала.

С учетом выделенных факторов влияния на восприятие информации в мобильном

приложении был разработан стимульный материал, который содержит в себе факторы в различных комбинациях. Первый фактор – цветовое решение. Для стимульного материала были выбраны следующие цветовые сочетания: черно-белый, бело-черный, красно-зеленый, зелено-красный, сине-оранжевый и оранжево-синий. Примеры стимульного материала в различных цветовых решениях представлены на рисунке 3.



Рис. 3 – Фактор «Цветовое решение»

Цветовое решение стимульного материала выбиралось исходя из теории цветовых контрастов И. Иттена [\[20\]](#) и схемы цветового зрения Ф. Юрьева (Рис. 4) [\[21\]](#).

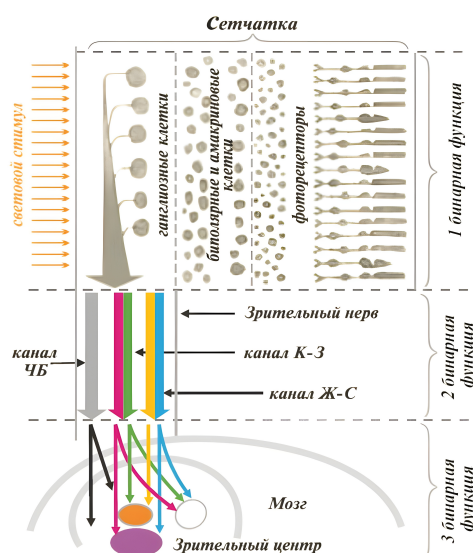


Рис. 4 – Схема цветового зрения человека

Второй фактор – композиционное решение, его основными элементами выступают квадратные плитки трех размеров. Данный фактор был определен в виде размера активных элементов: маленький, средний и крупный (Рис. 5). Они представлены в виде 6 равных квадратных плиток с расположенными на них управляющими словами и четырех иконок быстрого доступа. Размеры квадратных плиток 105*105px, 140*140px, 175*175px (разница между размерами - 35px). Они были выбраны с учетом размеров экрана смартфона. Такое композиционное решение было выбрано, исходя из проведенного предварительного исследования [\[8\]](#), и представляет собой доработанную методику проведения эксперимента.

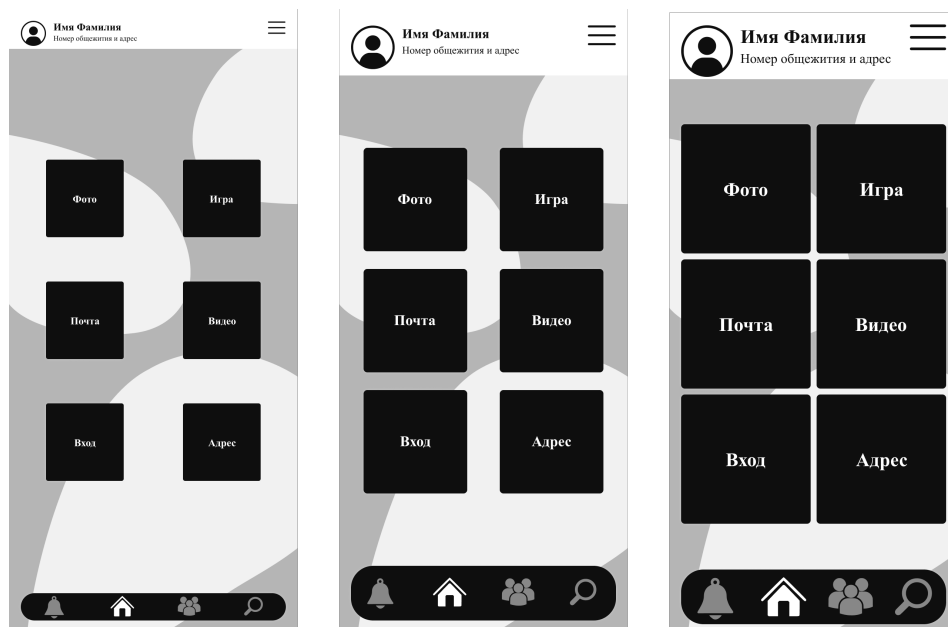


Рис. 5 – Фактор «Композиционное решение»

Третий фактор – тип шрифта. Для создания модели мобильного приложения были выбраны 2 шрифта – с засечками и без засечек. Шрифт с засечками – «Times New Roman», без засечек – «PT Sans» (Рис. 6).

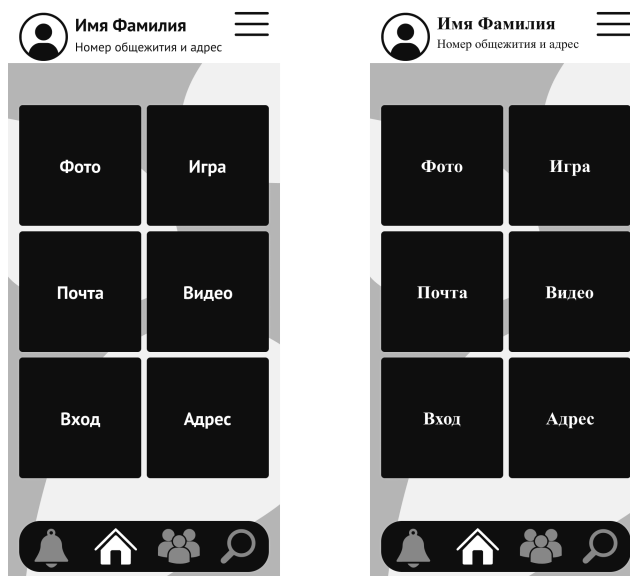


Рис. 6 – Фактор «Шрифт»

Четвертый фактор – количество букв в управляющих словах. Для проведения исследования были выбраны слова, которые используются в мобильных приложениях. Мобильные приложения делятся по различным категориям, например, мессенджеры, финансы, музыка, соц. сети, исходя из того, для какой аудитории оно разрабатывается и какие цели преследует [\[7\]](#). Каждой категории мобильных приложений характерны собственные технические свойства и особенности, также как и управляющие слова. Используемые слова отличаются между собой по количеству букв. Короткие управляющие слова состоят из 4-5 букв, а длинные из 9-11 букв (Рис. 7).

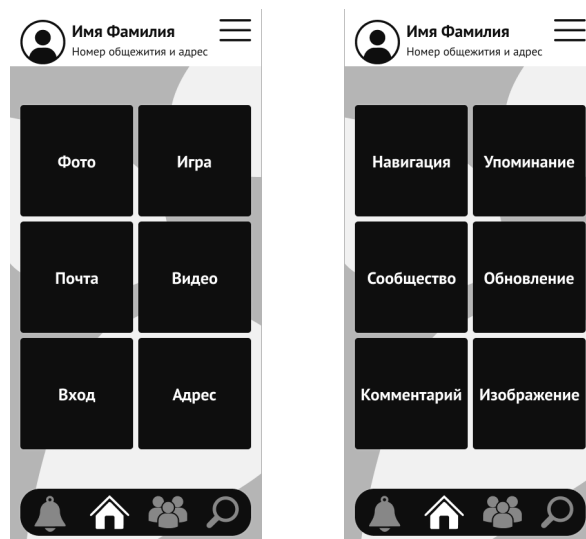


Рис. 7 – Фактор «Количество букв в управляющих словах»

Для проведения эксперимента было разработано 72 стимула с различным сочетанием исследуемых факторов. Для испытуемых была определена следующая задача: «Посмотрите на иконку и нажмите на плитку». После нажатия на нужный элемент в стимуле испытуемый автоматически переходил к следующему стимулу. Задача эксперимента позволяет проанализировать эффективность решения испытуемым поставленных задач в зависимости от цветового решения, композиционного решения, шрифта и количества букв в управляющих словах в мобильном приложении. Отображение задачи в программном экспериментальном модуле представлено на рисунке 8.



Рис. 8 – Задача для испытуемых в эксперименте

Эксперимент был проведен в специализированном программно-аппаратном комплексе с технологией ай-трекинга SMI RED 250 [22]. Для проведения исследования было выделено помещение, в котором была зафиксирована экспериментальная установка, включающая в себя ай-трекер, монитор с разрешением экрана 1920*1080 px и размером экрана 450*280 мм, стол, офисное кресло и офтальмологическую подставку под подбородок для фиксации головы участников эксперимента.

В исследовании приняли участие 36 человек – 20 женщин и 16 мужчин в возрасте от 20 до 30 лет из числа студентов и преподавателей Санкт-Петербургского политехнического университета Петра Великого. Участники эксперимента были разбиты на группы по следующим признакам:

- гендерный признак (мужской/женский);
- тип образования (гуманитарное/техническое);
- наличие художественной подготовки (имеется/не имеется).

Каждый фактор, влияющий на восприятие информации в мобильном приложении, отображается в заданной системе кодирования:

1. Цветовое решение: bW – бело-черное, wB – черно-белое, rG – красно-зеленое, gR – зелено-красное, oB – оранжево-синее, bO – сине-оранжевое, где строчная буква представляет собой цвет плашек, а прописная цвет фона;
2. Композиционное решение: b – большие квадратные плитки, m – средние квадратные плитки, s – маленькие квадратные плитки;
3. Шрифт: z – с засечками, n – без засечек;
4. Количество букв в управляющих словах: ln – слово из 9-11 букв, sh – слово из 4-5 букв;
5. Пол: 0 – женский, 1 – мужской;
6. Образование: 0 – гуманитарное, 1 – техническое;
7. Художественная подготовка: 0 – не имеется, 1 – имеется.

Всего в эксперименте было собрано 18543 фиксации и 16810 саккад. Результаты эксперимента были обработаны посредством программной процедуры дисперсионного анализа ANOVA. Данная процедура использует стандартные статистические методы и встроена во все автоматизированные системы статистической обработки экспериментальных данных [23]. Для анализа результатов эксперимента использовалось автоматизированный комплекс обработки экспериментальных данных STATISTICA [24]. Уровень значимости p-value был принят равным 0.05. Результаты исследования представлены на рисунках далее.

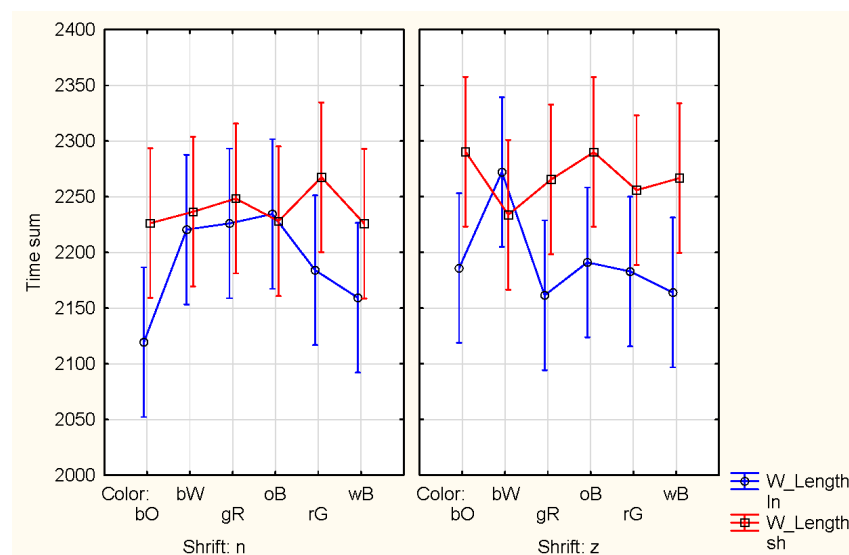


Рис. 9 – Плотность распределения времени рассматривания стимула в зависимости от шрифта, количества букв в слове и цветового решения

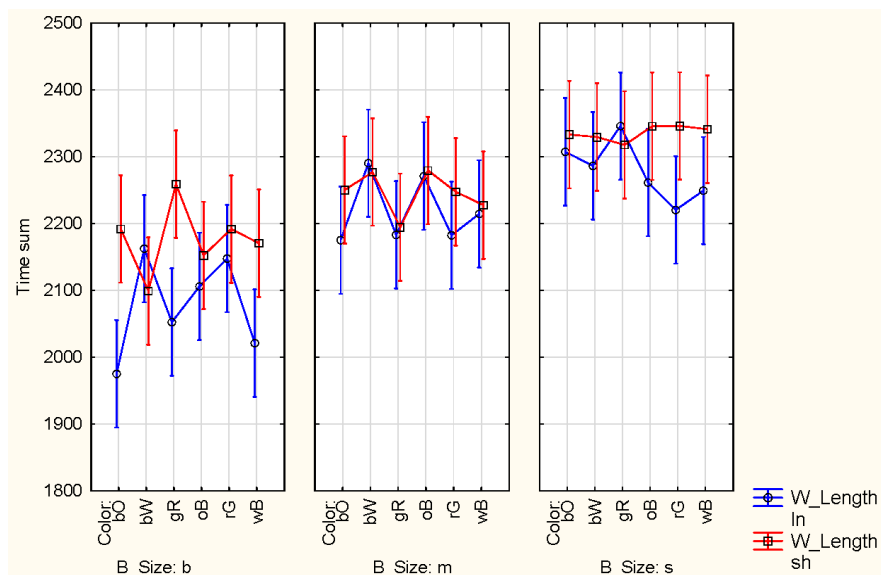


Рис. 10 – Плотность распределения времени рассматривания стимула в зависимости от композиционного решения, количества букв в слове и цветового решения

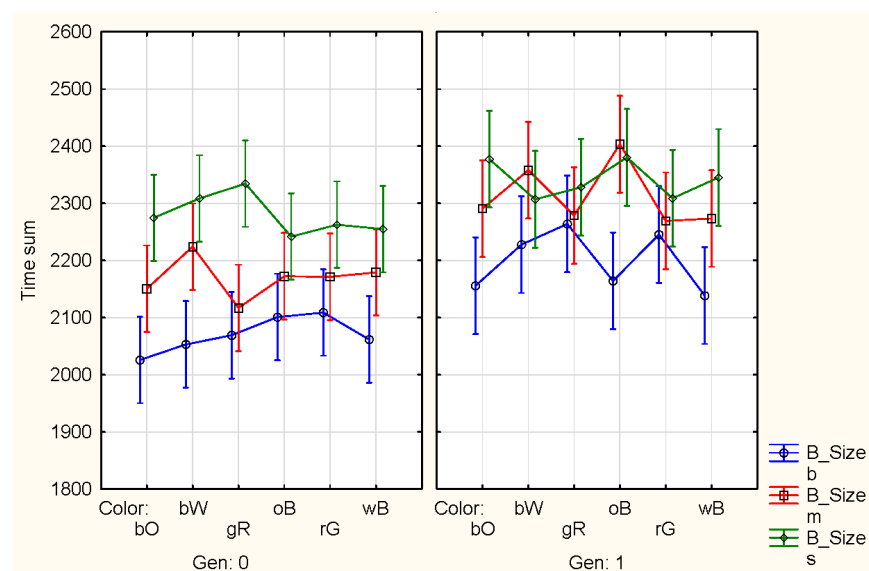


Рис. 11 – Плотность распределения времени рассматривания стимула в зависимости от пола, композиционного решения и цветового решения

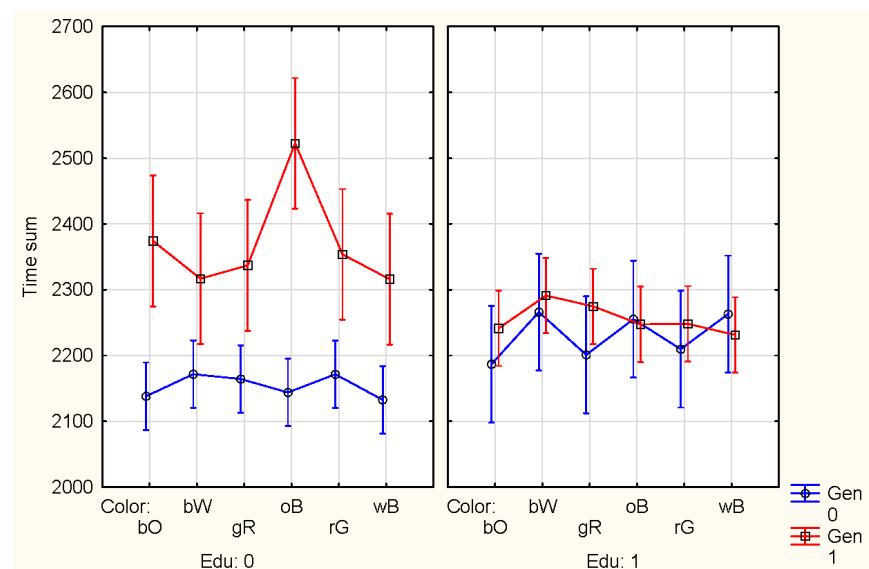


Рис. 12 – Плотность распределения времени рассматривания стимула в зависимости от

пола, образования и цветового решения

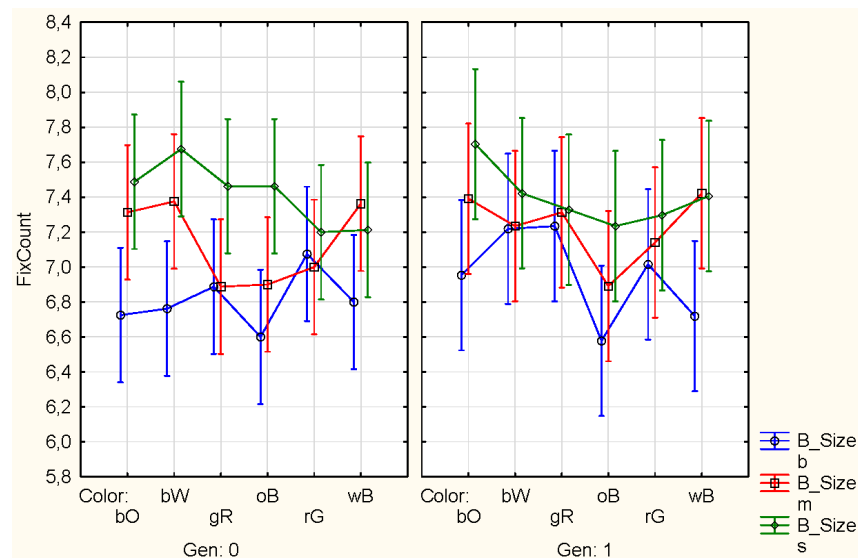


Рис. 13 – Плотность распределения количества фиксаций при рассматривании стимула в зависимости от пола, композиционного решения и цветового решения

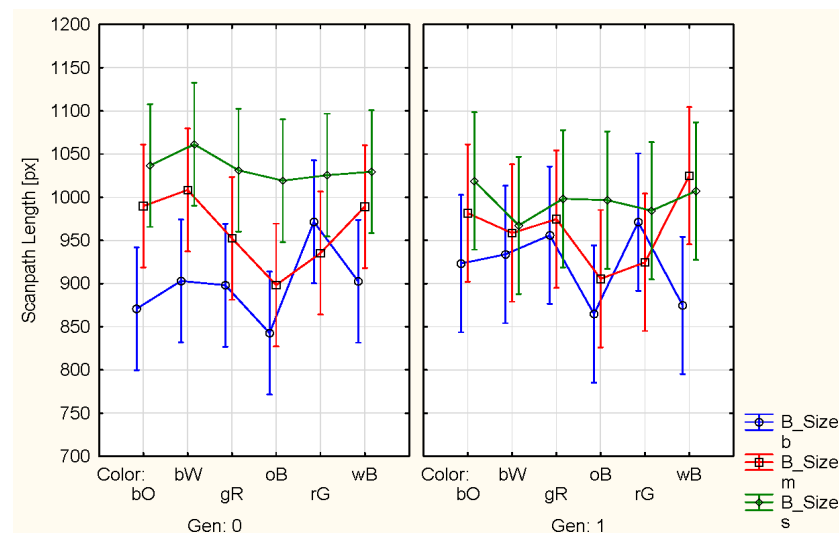


Рис. 14 – Плотность распределения расстояния пройденными саккадами при рассматривании стимула в зависимости от пола, композиционного решения и цветового решения

Шесть представленных графиков (9,10,11,12,13,14) демонстрируют, что быстрее всего участники эксперимента решали задачу в сине-оранжевом и бело-черном цветовых сочетаниях стимульного материала, а также наиболее быстро испытуемые решали задачи с крупными активными элементами в композиционном решении.

Кроме того, эксперимент показал, что такие выделенные категории, как образование (техническое/гуманитарное), пол (мужской/женский), художественная подготовка (имеется/не имеется) тоже имеют статистическое влияние на восприятие информации в мобильном приложении. Женщины решали поставленные задачи быстрее мужчин. Женщины с гуманитарным образованием решали поставленные задачи быстрее мужчин с гуманитарным образованием. Женщины и мужчины с техническим образованием справились с представленными задачами одинаково. Проследивание влияния пола (Gen) и образования (Edu) подтверждается данными, собранными при помощи процедуры ANOVA. Было выявлено, что факторы: «Пол» (Gen) и «Образование» (Edu) меньше 0.05.

Это позволяет предположить, что указанные факторы влияют на решение поставленной в эксперименте задачи (Таблица 1).

Таблица 1 – Значения p-value, полученные в результате выполнения программной процедуры дисперсионного анализа ANOVA. Факторы: Color, Gen, Edu

Фактор	P-value
Color	0,293
Gen	0,000001
Edu	0,238
Color*Edu	0,379
Color*Gen	0,3
Gen*Edu	0,000001

Далее мы увидели, что проявляется статистически значимое влияние шрифта с засечками. При его использовании в стимульном материале испытуемые решают задачу эксперимента быстрее. Интересным наблюдением стало то, что длинные слова воспринимались испытуемыми быстрее коротких, это можно увидеть в статистических данных процедуры ANOVA: фактор «количество букв в управляющих словах» (W_Length) меньше 0.05, поэтому можно предположить, что данный фактор влияет на решение поставленной в эксперименте задачи (Таблица 2).

Таблица 2 – Значения p-value, полученные в результате выполнения программной процедуры дисперсионного анализа ANOVA. Факторы: Color, W_Length, Shrift

Фактор	P-value
Color	0,559
W_Length	0,000013
Shrift	0,275

Таким образом, все выделенные факторы: цветовое решение, композиционное решение, шрифт и количество букв в управляющих словах влияют на восприятие в мобильном приложении, но при определенных условиях.

Полученные результаты позволяют разработать следующие рекомендации для разработки мобильного приложения:

1. Светлая и темная темы, в которых будут отображаться различные шрифты: светлая тема – шрифт без засечек, темная тема – шрифт с засечками;
2. Элементы мобильного приложения в черно-белом и сине-оранжевом цветовых сочетаниях;
3. Длинные слова в элементах мобильного приложения;
4. Крупные квадратные плитки, расположенные на экранах мобильного приложения;
5. Разделение мобильного приложения по половому признаку.

Данные рекомендации могут быть дополнены и расширены благодаря проведению дополнительных исследований с помощью технологии ай-трекинга.

Результаты работы:

1. Разработанная методика продемонстрировала свою работоспособность;

2. Данная методика может быть использована для тестирования мобильных приложений;
3. Данная методика позволяет разработать мобильное приложение с учетом выделенных рекомендаций;
4. Необходимо разрабатывать следующие задачи для проведения новых экспериментов.

Апробация разработанной методики прошла успешно, были выделены рекомендации, которые могут помочь при разработке мобильного приложения. Кроме того, методика показала результаты, которые требуют проведения дополнительных, более глубоких исследований.

Библиография

1. Морева Д.В., Камальдинова З.Ф. Особенности тестирования мобильных приложений // Цифровые технологии: настоящее и будущее. 2022. С. 87-92.
2. Заяц А.М., Васильев Н.П. Введение в гибридные технологии разработки мобильных приложений. СПб.: «Лань», 2022.
3. Сугаипов С.А.А., Гериханов З.А. Искусство разработки мобильных приложений // Тенденции развития науки и образования. 2023. № 98-10. С. 134-136.
4. Пантелейкин Н.В. Мобильные приложения и их виды // Научно-методический электронный журнал «Концепт». 2016. Т. 26. С. 776-780.
5. Богданова В.С. Принципы разработки интерфейса мобильного приложения // Наука и производство Урала. 2021. Т. 17. С. 108-110.
6. Плетнева А.О., Янчус В.Э. Разработка методики тестирования мобильного приложения на основе технологии ай-трекинга // Труды международной конференции по компьютерной графике и зрению «Графикон». 2023. № 33. С. 959-968.
7. Гиматдинов Р.С. Тенденции развития сферы разработки мобильных приложений // Вопросы студенческой науки. 2019. № 6(34). С. 359-362.
8. Овчинников М.А., Овчинникова М.А. Оценка качества пользовательского интерфейса при помощи технологии отслеживания взгляда // Современные информационные технологии в образовании, науке и промышленности. 2023. С. 75-77.
9. Лаптев М.В., Янчус В.Э., Лаптев В.В. Выявление зон интереса пользователя при визуализации данных с использованием метода ай-трекинга // ООО «ПСК "ФРАМ"». 2023. С. 49-62.
10. Хейфиц А.Е., Янчус В.Э., Борович Е.В. Методика проведения экспериментального исследования восприятия графического интерфейса с применением технологии ай-трекинга // Программные системы и вычислительные методы. 2022. № 2. С. 52-62.
11. Сарычева Ю.Ю., Федоров В.О. Применение устройств отслеживания взгляда при исследовании интерфейсов программных продуктов // Научно-практический электронный журнал. 2022. Т. 12. № 8. С. 52-63.
12. Самогоров В.А., Конкина Е.Д. Теория цветовых контрастов Йоханнеса Иттена // Самарский вестник. 2021. Т. 11. № 3. С. 97-103.
13. Янчус В.Э., Борович Е.В. Исследование значения цветового решения в процессе гармонизации кинокадра // Научно-технические ведомости СПбГПУ. 2016. №4. С. 53-68.
14. Орлов П.А., Лаптев В.В., Иванов В.М. К вопросу о применении систем ай-трекинга // Научно-технические ведомости СПбГПУ. 2014. №5(205). С. 84-94.
15. Шеффе Г. Дисперсионный анализ. Москва: Наука, 1980.
16. Боровиков В.П. Statistica: искусство анализа данных на компьютере. СПб: Питер, 2003.

Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Рассмотренная статья посвящена разработке методики тестирования мобильных приложений с использованием технологии ай-трекинга. Авторы акцентируют внимание на том, что современные пользователи активно взаимодействуют с различными гаджетами, в частности с мобильными телефонами, которые используются для широкого спектра задач. Смартфоны, благодаря своим функциональным возможностям, стали незаменимыми помощниками в повседневной жизни, обеспечивая доступ к Интернету, мобильной связи, социальным сетям, мессенджерам и множеству приложений. Учитывая малые размеры устройств и большой объем информации, которую необходимо обрабатывать, разработка мобильных приложений является значимой задачей для многих компаний.

Авторы подчеркивают важность проектирования мобильных приложений с учетом узнаваемых пользователем элементов, обеспечивая их очевидность, понятность и предсказуемость в процессе взаимодействия. Основные элементы дизайна включают цветовое решение, композицию и выбор шрифта, при этом особое внимание уделяется четкости, понятности и структурированности текста, что критически важно в условиях различных сценариев использования смартфонов.

Целью исследования является разработка методики исследования восприятия визуальной информации в мобильном приложении, для чего авторами были поставлены задачи по разработке стимульного материала, созданию задачи для испытуемых, проведению эксперимента в специализированном программном модуле и анализу результатов с помощью математической статистики. Гипотеза исследования заключается в том, что различные факторы, такие как цветовое и композиционное решение, шрифт и количество букв в управляющих словах, оказывают влияние на скорость решения задачи испытуемыми при взаимодействии с графическим интерфейсом мобильного приложения.

Для исследования был использован программно-аппаратный комплекс ай-трекинга, который позволяет записывать координаты взгляда пользователя и затем анализировать полученные данные для выявления поведенческих особенностей, незаметных для пользователя. Эти данные подвергаются статистическому анализу для выявления влияния различных факторов на шаблон восприятия информации.

В исследовании были выделены и тщательно изучены следующие факторы, влияющие на восприятие информации в мобильных приложениях:

1. Цветовое решение: использовались различные цветовые сочетания на основе теории цветовых контрастов И. Иттена и схемы цветового зрения Ф. Юрьева. Эти сочетания включали черно-белое, бело-черное, красно-зеленое, зелено-красное, сине-оранжевое и оранжево-синее.
2. Композиционное решение: рассматривались квадратные плитки трех различных размеров, расположенные на экране, чтобы изучить, как размер активных элементов влияет на восприятие.
3. Тип шрифта: использовались шрифты с засечками (например, Times New Roman) и без засечек (например, PT Sans) для оценки их влияния на удобство чтения и восприятие информации.
4. Количество букв в управляющих словах: анализировалось, как длина слов, используемых в интерфейсе, влияет на понимание и удобство использования приложения.

Считаю, что статья представляет собой ценный вклад в область разработки мобильных приложений, демонстрируя важность учета различных факторов, таких как цветовое и композиционное решение, выбор шрифта и длина слов, влияющих на восприятие пользователя. Результаты исследования, подкрепленные статистически значимыми данными, позволяют сформулировать конкретные рекомендации для разработчиков, направленные на повышение удобства и эффективности мобильных приложений.

Применение шрифта с засечками в темной теме и шрифта без засечек в светлой теме, а также использование черно-белых и сине-оранжевых цветовых сочетаний, могут значительно улучшить пользовательский опыт. Кроме того, включение длинных слов и крупных квадратных плиток в дизайн может способствовать более быстрому восприятию и обработке информации пользователем.

Особенно заслуживает внимания рекомендация по разделению приложения по половому признаку, что может открыть новые перспективы для персонализации пользовательского интерфейса. В целом, предложенные авторами рекомендации отражают глубокое понимание того, как детали дизайна могут влиять на функциональность и восприятие мобильных приложений, и предоставляют полезный инструментарий для разработчиков в этой области.

Англоязычные метаданные

Using computer simulation of the hand movement of an anthropomorphic robot to determine the position of its base relative to objects of manipulation

Pritykin Fedor Nikolaevich 

Doctor of Technical Science

Professor, Omsk State Technical University

644050, Russia, Omskaya oblast', g. Omsk, ul. Prospekt Mira, 11, aud. Zv-516

✉ pritykin@mail.ru

Abstract. The subject of the study is the use of an intelligent hand movement control system of an anthropomorphic robot when performing operations of placing objects of manipulation in a container of specified dimensions. The object of the study is the process of determining the parameters of the relative position of the base of the hand in relation to the objects of manipulation and the container, in which it is possible to perform motor tasks. The author examines in detail the algorithm for determining the position of the base of the arm of an anthropomorphic robot using the synthesis of movements along the velocity vector when solving the problem of installing manipulating objects specified in the form of rectangular prisms into a container. Special attention is paid in the article to the methodology of determining the center of the coordinate system associated with the base of the robot in a fixed space and calculating the target points to which the center of the output link moves at various positions of previously installed manipulation objects in the container. The essence of the method is to use computer simulation of the movement of an anthropomorphic robot using the synthesis of movements along the velocity vector with an assessment of the relative position of the arm mechanism and restricted areas. Previously installed manipulation objects located inside the container and the side walls of the container itself act as restricted areas. The main conclusions of the presented study are the possibility of using the developed algorithm to test the synthesis of hand movements with given geometric parameters that set the position of an anthropomorphic robot, conveyor and container, in which there is no occurrence of deadlocks. The novelty of the research consists in the development of a method based on an iterative search for the values of the parameters of the mutual position of the base of the arm, conveyor and container at each iteration in case of deadlock situations. The results of calculations of the position of the base of the robot arm and intermediate configurations constructed using computer simulation of movements based on the use of the developed algorithm are presented. The conducted research can be used in the development of information and control complexes of mobile objects, in particular in the development of intelligent control systems for autonomously functioning anthropomorphic robots in organized environments.

Keywords: objects of manipulation, kinematic chain model, deadlocks, anthropomorphic robots, movement synthesis, computer modeling, geometric modeling, manipulator mechanisms, intelligent control systems, output link

References (transliterated)

1. Shcherbatov I. A. Intellektual'noe upravlenie robototekhnicheskimi sistemami v usloviyakh neopredelennosti // Vestnik AGTU, ser. Upravlenie, vychislitel'naya tekhnika

- i informatika. 2010. № 1. S. 73-77.
2. Yushchenko A. S. Intellectual'noe planirovanie v deyatel'nosti robotov // Mekhatronika, avtomatizatsiya, upravlenie. 2005. № 3. S. 5-18.
 3. Afonin V.L., Bazrov B.M., Sleptsov V.V. Tenzornoe ischislenie v zadachakh upravleniya i predstavleniya znaniy v intellektual'nykh obrabatyvayushchikh tekhnologicheskikh sistemakh // Problemy mashinostroeniya i avtomatizatsii. 2021. № 1. S. 9-18.
 4. Melekhin V.B., Khachumov M.V. Planirovanie tselenapravlennoi deyatel'nosti avtonomnym intellektual'nym robotom s obnovleniem znaniy v kratkovremennoi pamyati // Mekhatronika, avtomatizatsiya, upravlenie. 2024. T. 25. № 2. S. 79-92.
 5. Melekhin V.B., Khachumov M.V. Kognitivnye instrumenty abstraktnogo myshleniya avtonomnykh intellektual'nykh mobil'nykh sistem // Mekhatronika, avtomatizatsiya, upravlenie. 2023. T. 24, № 6. S. 317-326.
 6. Man'ko S.V., Lokhin V.M., Krainov N.V., Mal'ko A.N. Algoritmy intellektual'nogo upravleniya rekonfiguriruemyimi robotami v komponovke koleasa i mnogoagentnymi sistemami na ikh osnove // Mekhatronika, avtomatizatsiya, upravlenie. 2022. T. 23, № 8. S. 420-429.
 7. Dubenko Yu. V., Dyshkant E. E., Timchenko N. N., Rudeshko N. A. Gibridnyi algoritm formirovaniya kratchaishei traektorii, osnovannyi na primenenii mnogoagentnogo obucheniya s podkrepleniem i obmena opytom // Vestnik komp'yuternykh i informatsionnykh tekhnologii. 2021. № 11. S. 13-26.
 8. Kutlubaev, I.M. Bogdanov, A.A., Novoseltsev, N.V., Krasnobaev, M.V., Saprykin, O.A. (2016) Control system of the anthropomorphous robot for work on the low-altitude earth orbit. *International Journal of Pharmacy and Technology*, 3, 18913-18199.
 9. Hasegawa, T., Suehiro, T., Takase, K. (1992) A model-based manipulation system with skill-based xecution. *IEEE Trans. Rob. and Autom.*, vol. 8, 535-544.
 10. Pratt, J. , Dilworth P., Pratt, G. (1997) Virtual model control of a bipedal walking robot. *Proceedings of Int. Conf. on Robotics and Automation*, vol. 1, 193-198.
 11. Hrr, J. Pratt, J.Chew, C.Herr, H.Pratt, G. (1998) Adaptive virtual model control of a bipedal walking robot. *IEEE Inter. Simps. Intellegence and Systems*, 245-251.
 12. Tsukamoto, H., Takubo, T. ,Ohara, K., Mae, Y., Arai, T. (2010) Virtual impedance model for obstacle avoidance in a limb mechanism robot. *(ICIA): IEEE Inf. and Autom.*, 729-734.
 13. Korendyasev A. I., Salamandra B. L., Tyves L. I. Manipulyatsionnye sistemy robotov. M: Mashinostroenie, 1989.
 14. Pritykin F. N. Virtual'noe modelirovanie dvizhenii robotov, imeyushchikh razlichnuyu strukturu kinematcheskikh tsepei. Omsk: Izd-vo OmGTU, 2014.
 15. Whitney, D. E. (2010) The Mathematics of Coordinated Control of Prosthetic Arms and Manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 94 (4), 303-309.
 16. Pritykin, F. N., Nebritov, V. I. (2021) Using weight values of generalized velocities to handle deadlocks in the synthesis of anthropomorphic robot arm movement. *Journal of Physics: Conference Series*, 1791(1), 012059.
 17. Pritykin, F. N., Nebritov, V. I. (2019) Determination of target points approachability by an android robot arm in organized space based on virtual modeling of movements . *Journal of Physics: Conference Series*, 1260, 072015.
 18. Pritykin F. N., Nebritov V. I. Graficheskaya optimizatsionnaya model' protsessa svarki izdelii robotom na chertezhe Radishcheva // Programmnye sistemy i vychislitel'nye metody. 2021. № 2. S. 63-73.

19. Pritykin F. N. Modelirovanie giperpoverkhnostei, otrazhayushchikh vzaimosvyaz' kinematicheskikh parametrov mekhanizma androidnogo robota // Vestnik komp'yuternykh i informatsionnykh tekhnologii. 2022. T. 19, № 7. С. 21-29.
20. Pritykin F. N. Issledovanie kinematicheskikh parametrov androidnogo robota pri avtomatizirovannom sinteze dvizhenii po vektoru skorosti // Omskii nauchnyi vestnik. 2022. № 2 (182). S. 5-9.

The impact of asynchronous and multithreaded query processing models on the performance of server-side web applications

Makarov Igor Sergeevich 

PhD in Technical Science

Head of the Department; Department of Applied Informatics (PI); Volga State University of Telecommunications and Informatics

23 Lva Tolstogo str., Samara region, 443010, Russia

✉ igor-psati@yandex.ru

Larin Denis Vyacheslavovich 

Student; Department of Computer Science and Computer Engineering; Volga Region State University of Telecommunications and Informatics

23 Lva Tolstogo str., Samara, Samara region, 443010, Russia

✉ denlar1989@gmail.com

Vorobeva Evgeniia Grigor'evna 

Student; Department of Computer Science and Computer Engineering; Volga Region State University of Telecommunications and Informatics

23 Lva Tolstogo str., Samara region, 443010, Russia

✉ vorobeva.g2004@gmail.com

Emelin Daniil Pavlovich 

Student; Department of Computer Science and Computer Engineering; Volga Region State University of Telecommunications and Informatics

23 Lva Tolstogo str., Samara, Samara region, 443010, Russia

✉ demelin163@gmail.com

Kartashov Dmitry Aleksandrovich 

Student; Department of Computer Science and Computer Engineering; Volga Region State University of Telecommunications and Informatics

23 Lva Tolstogo str., Samara region, 443010, Russia

✉ tawerka40@gmail.com

Abstract. The object of the study is server-side web applications and their performance when processing a large number of simultaneous requests. Asynchronous technologies (Node.js, Python Asyncio, Go, Kotlin Coroutines) and multithreaded models (Java Threading, Python Threading). The authors analyze asynchronous event loops, goroutines, coroutines, and classical multithreaded approaches in detail, evaluating their effectiveness in tasks with intensive use of I/O and computing resources. An experiment is underway with API development in three languages (Java, Node.js, Go) and testing using the hey utility. It also explores the features of scalability, performance optimization, caching, error handling, load tests, and implementation features of parallel computing. The purpose of the study is to

determine which approaches provide the highest performance in server applications. Research methods include load testing, collection of metrics (response time, bandwidth, and server resource consumption), and analysis of the results. The scientific novelty lies in comparing asynchronous and multithreaded methods in real-world web development scenarios. The main conclusions of the study are recommendations on the use of asynchronous technologies in high-load I/O tasks and multithreading in computationally complex scenarios. The results obtained will help developers optimize the performance of server applications depending on their tasks and workload. Additionally, the study examines aspects of the complexity of debugging asynchronous applications, the impact of thread pools on the performance of multithreaded solutions, as well as scenarios in which asynchronous and multithreaded approaches can complement each other. Special attention is paid to server resource management under scalable loads, which will allow IT specialists to more accurately select tools and technologies for solving specific tasks. In conclusion, possible ways to optimize the operation of server applications are discussed, including the use of new approaches and algorithms, as well as the prospects for the development of asynchronous and multithreaded technologies in the context of highly loaded systems, their impact on the overall application architecture, as well as on increasing fault tolerance and security.

Keywords: Go goroutines, Python Asyncio, load testing, Node.js, server-side web applications, efficiency, Java Threading, multithreading, asynchrony, performance of server applications

References (transliterated)

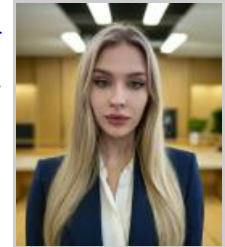
1. Opivalov S. A. Metody raboty s potokami v yazyke Java // Mezhdunarodnyi zhurnal gumanitarnykh i estestvennykh nauk. 2023. № 4 (79) T. 3. S. 93-99.
2. Rukovodstvo po Node.js, chast' 1: obshchie svedeniya i nachalo raboty [Elektronnyi resurs]. URL: <https://habr.com/ru/companies/ruvds/articles/422893/> (Data obrashcheniya: 03.03.2025).
3. Opivalov S. A. Perspektivy ispol'zovaniya yazyka Kotlin v programirovanii // Mezhdunarodnyi zhurnal gumanitarnykh i estestvennykh nauk. 2023. № 9 (84) T. 1. S. 260-262.
4. Parallel'noe programirovanie v Go [Elektronnyi resurs]. URL: <https://proglib.io/p/parallelnoe-programirovanie-v-go-2021-05-23?ysclid=m7wag43tb6712881695> (Data obrashcheniya: 03.03.2025).
5. Asynchronous Functions and the Node.js Event Loop [Elektronnyi resurs]. URL: https://translated.turbopages.org/proxy_u/en-ru.ru.137d7d27-67c89218-0c37f6b9-74722d776562/https/www.geeksforgeeks.org/asynchronous-functions-and-the-node-js-event-loop/ (Data obrashcheniya: 04.03.2025).
6. Load Testing using Hey [Elektronnyi resurs]. URL: <https://dev.to/saantoryuu/load-testing-using-hey-c84> (Data obrashcheniya: 04.03.2025).
7. Process Explorer v17.06 [Elektronnyi resurs]. URL: <https://learn.microsoft.com/en-us/sysinternals/downloads/process-explorer> (Data obrashcheniya: 02.03.2025).

Comparative analysis of indexing strategies in PostgreSQL under various load scenarios

Independent researcher

394062, Russia, Voronezh region, Voronezh, lane Antokolsky, 4

✉ dar.zolott@gmail.com



Abstract. The subject of the study is the effectiveness of various indexing strategies implemented in PostgreSQL and their impact on the performance of SELECT, UPDATE and INSERT operations in conditions of different data scales. The object of the study are the B-Tree, GIN and BRIN indexes used to optimize the operation of databases. The author examines in detail such aspects of the topic as the time characteristics of operations, the size of indexes and their resource intensity. Special attention is paid to the impact of data volume on the performance of indexes and their suitability for working with various types of data, including JSONB. The research is aimed at systematizing knowledge about the use of indexes to improve the efficiency of highly loaded systems, which require optimization of access operations, updating and inserting data, as well as analysis of resource consumption. The leading research method is an empirical approach, which includes the development of a test database with orders, customers, and products tables. Experiments were conducted for SELECT, UPDATE, and INSERT operations on small, medium, and large amounts of data. The metrics of query execution time and index size obtained using PostgreSQL tools were used for the analysis. The novelty of the research lies in conducting a comprehensive performance analysis of the B-Tree, GIN, and BRIN indexes in PostgreSQL when performing typical operations on various amounts of data. The main conclusions of the conducted research are recommendations on the choice of indexes depending on the types of queries and their execution conditions. B-Tree indexes have proven their versatility, demonstrating high performance for SELECT and UPDATE operations. GIN indexes have shown advantages for working with JSONB data, but their use is limited by high resource intensity. BRIN indexes have proven to be effective for large amounts of data, especially for SELECT operations, due to their compactness and low overhead. A special contribution of the author to the research of the topic is the creation of recommendations for database developers, which makes it possible to improve application performance by choosing an optimal indexing strategy.

Keywords: data management, high-load systems, JSONB data, relational databases, PostgreSQL performance, query optimization, GIN indexes, B-Tree indexes, BRIN indexes, indexing strategies

References (transliterated)

1. Dombrovskaya G. Optimizatsiya zaprosov v PostgreSQL. M.: DMK-Press, 2022.
2. Mostafa A. S. A Case Study on B-Tree Database Indexing Technique // Journal of Soft Computing and Data Mining. 2020. № 27-3. URL: <https://penerbit.uthm.edu.my/ojs/index.php/jscdm/article/view/6828> (data obrashcheniya: 10.01.2025).
3. Borodin A., Mirvoda S., Porshnev S., Ponomareva O. Improving generalized inverted index lock wait times // Journal of Physics: Conf. 2018. №944. URL: <https://iopscience.iop.org/article/10.1088/1742-6596/944/1/012022/pdf> (data obrashcheniya: 10.01.2025).
4. Borodin A., Mirvoda S., Kulikov I., Porshnev S. Optimization of Memory Operations in Generalized Search Trees of PostgreSQL // Communications in Computer and

- Information Science. 2017. № 716. URL:
https://link.springer.com/chapter/10.1007/978-3-319-58274-0_19 (data obrashcheniya: 10.01.2025).
5. Rogov E. V. PostgreSQL 16 iznutri. M.: DMK Press, 2024.
 6. Morozov S. V., Nesterov S. A. Sravnitel'nyi analiz tipov indeksov v SUBD SQL Server i PostgreSQL // SAEC. 2024. № 2. S. 485–491.
 7. Selivanov E. O. Sravnenie tipov indeksov v razlichnykh sistemakh upravleniya bazami dannykh // Molodezh' i nauka: aktual'nye problemy fundamental'nykh i prikladnykh issledovaniy: Materialy VI Vserossiiskoi natsional'noi nauchnoi konferentsii molodykh uchenykh. Komsomol'sk-na-Amure: Komsomol'skii-na-Amure gosudarstvennyi universitet, 2023. S. 357–361.
 8. Kudashov A. S., Agapova V. A., D'yachkov D. A., Kazakova I. A. Obzor tipov indeksov i ikh primeneniye v sistemakh upravleniya bazami dannykh // Sovremennye tsifrovye tekhnologii. Materialy II Vserossiiskoi nauchno-prakticheskoi konferentsii. Barnaul: Altaiskii gosudarstvennyi tekhnicheskii universitet im. I. I. Polzunova, 2023. C. 299–303.
 9. Documentation PostgreSQL 15 – URL: <https://www.postgresql.org/docs/15/release-15-2.html> (date of access: 10.01.2025).
 10. Sorokin V. E. Khraneniye i effektivnaya obrabotka nechetkikh dannykh v SUBD PostgreSQL // Programmnye produkty i sistemy. 2017. № 4. URL: <https://app.amanote.com/v4.1.10/research/note-taking?resourceId=Pprz23MBKQvf0Bhi37cu> (data obrashcheniya: 10.01.2025).
 11. B-tree indeksy v bazakh dannykh na primere PostgreSQL. – URL: <https://techtrain.ru/talks/e2273ec8ca2b4ea692c65318a50c4be5> (data obrashcheniya: 10.01.2025).
 12. Medzhidov R. G. Analiz mnogokolonochnykh indeksov baz dannykh // Aktual'nye problemy prikladnoi matematiki, informatiki i mekhaniki : Sbornik trudov Mezhdunarodnoi nauchnoi konferentsii. Voronezh : Nauchno-issledovatel'skie publikatsii, 2019. C. 420–422.
 13. Bogatov I. V. Effektivnaya optimizatsiya zaprosov v SUBD Postgres / Akademicheskaya publitsistika. 2022. № 5-2. C. 59–64.

Screenshot testing as a multi-aspect type of automated dynamic verification for web applications

Makarov Konstantin Sergeevich 

PhD in Technical Science

Head of the Department; Department of Software and Information Systems Administration; Kursk State University

33 Radishcheva str., Kursk, Russia, 305000

✉ makarov_ks@kursksu.ru

Fatkin Ruslan Igorevich 

Postgraduate student; Department of Software and Information Systems Administration; Kursk State University

305014, Russia, Kursk region, Rybinovaya str., 26B, sq. 96

✉ ruslan4631@yandex.ru

Abstract. The subject of this study is multi-aspect screenshot testing as a modern method of

automated dynamic verification of web applications, combining functional testing and user interface (UI) validation. Contemporary testing methods face challenges such as high labor intensity, false positives, and low scalability, especially in complex projects. The main objective of the research is to develop and implement a method that improves defect detection accuracy, reduces testing time, and lowers test case development costs. The study explores image comparison algorithms, dynamic element filtering techniques, and automated UI analysis approaches to enhance efficiency and standardization in the web application verification process. Unlike functional and UI testing conducted separately, the proposed method enables simultaneous analysis of multiple aspects of the interface and functionality, minimizing labor costs and increasing testing reliability. The approach employs automated comparison of reference and test screenshots at the pixel, structural element, and content levels using Python, Selenium, PIL, and Pytest-xdist for parallel test execution, effectively addressing the challenges of web application verification. Some researchers in the field of testing agree that the testing process lacks standardization and clear evaluation criteria. The proposed method ensures the achievement of verification objectives even under evolving strategies and approaches to system performance assessment by creating a flexible and precise validation system that integrates various testing types into a unified structure, making it suitable for modern software development challenges. The experimental section demonstrates the advantages of multi-aspect screenshot testing over other methods, including reduced testing time, improved defect detection accuracy, and enhanced analysis of test reports. This approach can be adapted to various testing scenarios and is particularly beneficial for high-load projects requiring regular regression testing.

Keywords: testing criteria, testing optimization, UI testing, Selenium and PIL, Python in testing, dynamic verification web applications, multi-aspect screenshot testing, automated testing, screenshot testing, verification tasks

References (transliterated)

1. The Economic Impacts of Inadequate Infrastructure for Software Testing. NIST Report, May 2002. Rezhim dostupa: <https://www.nist.gov/system/files/documents/director/planning/report02-3.pdf> (Data obrashcheniya: 05.10.2024).
2. Gurin R. E., Rudakov I. V., Rebrikov A. V. Metody verifikatsii programmogo obespecheniya // Mashinostroenie i komp'yuternye tekhnologii. 2015. № 10. S. 235-251.
3. Quadri S. M. K., Farooq S. U. Software testing-goals, principles, and limitations // International Journal of Computer Applications. 2010. T. 6. № 9. S. 7-9.
4. Kumar S. Reviewing software testing models and optimization techniques: an analysis of efficiency and advancement needs // Journal of Computers, Mechanical and Management. 2023. T. 2. № 1. S. 43-55.
5. Xie Q., Memon A. M. Designing and comparing automated test oracles for GUI-based software applications // ACM Transactions on Software Engineering and Methodology (TOSEM). 2007. T. 16. № 1. S. 4.
6. Kudryavtseva E. Yu. Avtomatizirovannoe testirovanie veb-interfeisov // Gornyi informatsionno-analiticheskii byulleten' (nauchno-tekhnicheskii zhurnal). 2014. № S. S. 354-356.
7. Kulyamin V. V. Metody verifikatsii programmogo obespecheniya / V. V. Kulyamin. M.: ISP RAN, 2008. 111 s.

8. Persival' G. Python. Razrabotka na osnove testirovaniya / G. Persival'. M.: DMK Press, 2018. 622 s.
9. Beregeiko O. P., Dubovskii A. S. Avtomatizatsiya testirovaniya veb-prilozhenii // Vestnik magistratury. 2016. № 12-4 (63). S. 39-41.
10. Dwarakanath A., Neville D., Sanjay P. Machines that test Software like Humans. arXiv preprint arXiv:1809.09455 (2018).

Intelligent Monitoring and Route Adaptation System for Drones based on Neural Network Risk Analysis

Sergeev Dmitry Anatolyevich 

PhD in Economics

Associate Professor at the Graduate School of Industrial Economics; Peter the Great St. Petersburg Polytechnic University

50 Novorossiysk Street, Saint Petersburg, Russia, 194021

✉ sergeev_da@spbstu.ru

Rodionov Dmitry Grigoryevich 

Doctor of Economics

Professor; Director at the Graduate School of Industrial Economics; Peter the Great St. Petersburg Polytechnic University

50 Novorossiysk Street, Saint Petersburg, 194021, Russia

✉ drodionov@spbstu.ru

Polyakov Prohor Aleksandrovich 

Researcher at the Laboratory "Polytech-Invest"; Peter the Great St. Petersburg Polytechnic University

50 Novorossiysk Street, Saint Petersburg, 194021, Russia

✉ prohor@polyakov-box.ru

Golikov Gleb Igorevich 

Researcher at the Laboratory "Polytech-Invest"; Peter the Great St. Petersburg Polytechnic University

50 Novorossiysk Street, Saint Petersburg, 194021, Russia

✉ golikov.23@mail.ru

Starchenkova Olesya Dmitrievna 

Specialist at the Laboratory "Modeling and Digitalization of Socio-Economic Systems"; Peter the Great St. Petersburg Polytechnic University

50 Novorossiysk Street, Saint Petersburg, Russia, 194021

✉ starchenkova_od@spbstu.ru

Dmitriev Nikolai Dmitrievich 

PhD in Economics

Associate Professor at the Graduate School of Industrial Economics; Peter the Great St. Petersburg Polytechnic University

Head of the Laboratory "Modeling and Digitalization of Socio-Economic Systems"; Peter the Great St. Petersburg Polytechnic University

50 Novorossiysk Street, Saint Petersburg, 194021, Russia

✉ dmitriev_nd@spbstu.ru

Konnikov Evgenii Aleksandrovich 

Abstract. The study examines an intelligent monitoring and route adaptation system for unmanned aerial vehicles (Drones) based on neural network risk analysis. The research considers autonomous navigation algorithms that enable environmental analysis and real-time trajectory correction in response to potential threats. The possibilities of applying computer vision, neural network algorithms, data preprocessing methods, object detection, semantic segmentation, trajectory planning algorithms, predictive control, and adaptive route optimization are assessed for identifying obstacles, moving objects, and restricted flight zones. The role of intelligent control systems for drones is analyzed, along with their impact on enhancing autonomy, resilience, and operational efficiency in dynamically changing environments. The proposed solutions are aimed at risk mitigation associated with emergency situations through the implementation of adaptive flight control strategies. The study employs methods of systems analysis, computer vision, and machine learning, including convolutional neural networks, image preprocessing algorithms, data filtering and segmentation, as well as sensor data analysis. The effectiveness assessment is carried out through trajectory modeling, testing of threat identification algorithms, and analysis of UAV route stability parameters. The scientific novelty lies in the development of an integrated system for intelligent UAV route correction based on neural network object classification methods and adaptive trajectory planning algorithms. Mechanisms for predictive risk analysis have been developed, ensuring automatic route adjustment upon detecting obstacles, adverse weather conditions, and restricted-access zones. The proposed control architecture integrates computer vision technologies, data stream analysis, and automated decision-making while using dynamic routing methods, real-time flight correction algorithms, and collision avoidance strategies. This approach enhances the level of UAV autonomy. The developed intelligent navigation algorithms can be implemented in modern autonomous UAV control systems, enabling adaptation to dynamic conditions and increasing the efficiency of task execution across various domains, including defense and industrial applications.

Keywords: Unmanned Aerial Vehicles, Route Optimization, Threat Prevention, Autonomous Navigation, Deep Learning, Computer Vision, Neural Network Analysis, Intelligent Control, Decision-Making System, Adaptive Algorithms

References (transliterated)

1. Yakovlev A. G., Baranov D. E., Shishkin M. D. O sovremenom sostoyanii i tendentsiyakh primeneniya novykh materialov i tekhnologii v konstruktsiyakh bespilotnykh letatel'nykh apparatov // Vektor nauki Tol'yattinskogo gosudarstvennogo universiteta. 2014. № 1. S. 71-74.
2. Varlamov O. O. Ob odnom podkhode k metrike avtonomnosti i intellektual'nosti robototekhnicheskikh kompleksov // Izvestiya Kabardino-Balkarskogo nauchnogo tsentra RAN. 2017. № 6-2. S. 43-53.
3. Koptev V. A. Analiz BPLA kak aktual'nykh radiolokatsionnykh tselei i ikh maskiruyushchikh faktorov // Mezhdunarodnyi zhurnal gumanitarnykh i estestvennykh nauk. 2024. № 6-1. S. 244-250.

4. Varlamov O. O., Lazarev V. M., Chuvikov D. A., Dzhkha P. O perspektivakh sozdaniya avtonomnykh intellektual'nykh robotov na osnove mivarnykh tekhnologii // Radiopromyshlennost'. 2016. № 4. S. 96-105.
5. V Rossii razrabotali novye mnogofunktsional'nye drony «Kasatka» i «Mikrob» [Elektronnyi resurs]. URL: <https://aif.ru/society/army/v-rossii-razrabotali-novye-mnogofunktsionalnye-drony-kasatka-i-mikrob> (data obrashcheniya: 26.01.2025).
6. V Rossii predstavili novye mnogofunktsional'nye razrabotki dlya SVO [Elektronnyi resurs]. URL: <https://news.mail.ru/svo/64559772/> (data obrashcheniya: 26.01.2025).
7. Malyshev V. P., Vinogradov O. V. Vozmozhnye napravleniya povysheniya ustoichivosti organizatsii, neobkhodimykh dlya vyzhivaniya naseleniya pri sovremennykh voennykh konfliktakh // Problemy analiza riska. 2023. № 5. S. 60-70.
8. Korneev D. S. Ispol'zovanie apparata neironnykh setei dlya sozdaniya modeli otsenki i upravleniya riskami predpriyatiya // Upravlenie bol'shimi sistemami: sbornik trudov. 2007. № 17. S. 81-102.
9. Ovchinskii V. S. Tekhnologii budushchego protiv kriminala. M.: LitRes, 2019.
10. Belyakova E. V., Prokopovich D. A., Ryzhaya A. A., Gil'ts N. E. Logisticheskii podkhod k tekhnologicheskomu razvitiyu promyshlennogo kompleksa regiona // Vestnik Sibirskogo gosudarstvennogo aerokosmicheskogo universiteta im. akademika M.F. Reshetneva. 2015. № 4. S. 1007-1013.
11. Vyrelkin A. D., Kucheryavii A. E. Ispol'zovanie bespilotnykh letatel'nykh apparatov dlya resheniya zadach «umnogo goroda» // Informatsionnye tekhnologii i telekommunikatsii. 2017. № 1. S. 105-113.
12. Krinichko V. A., Knyazev E. A. Primenenie BPLA v boevoi rabote komandirov // Nauchnyi aspekt. 2023. № 8. S. 537-546.
13. Dovgal' V. A., Dovgal' D. V. Analiz bezopasnosti roya dronov, protivostoyashchego atakam zloumyshlennikov // Distsionnye obrazovatel'nye tekhnologii: sbornik trudov V Mezhdunarodnoi nauchno-prakticheskoi konferentsii. Simferopol': Arial, 2020. S. 372-377.
14. Lastochkin Yu. I., Yarygin Yu. N., Byvshikh D. M. Metodicheskoe obespechenie obosnovaniya sposobov boevogo primeneniya sil i sredstv radioelektronnoi bor'by pri protivodeistvii radioelektronnoi razvedke v operatsiyakh ob"edinenii sukhoputnykh voisk // Voennaya mysl'. 2018. № 6. S. 58-66.
15. Burchenkov V. V. Avtomatizirovannye sistemy kontrolya podvizhnogo sostava. Gomel': Belorusskii gosudarstvennyi universitet transporta, 2020. 226 s.
16. Borovik V. S., Gutsul V. I., Klestov S. A., Firsov I. S., Pykhtyrev V. S., Shatravin V. V., Syryamkin V. I., Shidlovskii S. V., Shashev D. V., Nguen K. T., Ugryumov D. A., Syryamkin M. V. Kollektivy intellektual'nykh robotov. Sfery primeneniya. Tomsk: STT, 2018. 140 s.
17. Terent'ev V. B. Metod svertki veroyatnostnykh i parametricheskikh pokazatelei letatel'nogo apparata // Fundamental'nye i prikladnye nauchnye issledovaniya: aktual'nye voprosy. 2019. S. 51.
18. Sorokin I. A., Romanov P. N., Chesnokov A. D., Kondranenkova T. E. Matematicheskaya model' obrabotki izobrazhenii opasnykh i vredonosnykh rastenii s kamery BPLA // Vestnik NGIEI. 2023. № 5. S. 20-32.
19. Ali B., Sadekov R. N., Tsodokova V. V. Algoritmy navigatsii bespilotnykh letatel'nykh apparatov s ispol'zovaniem sistem tekhnicheskogo zreniya // Giroskopiya i navigatsiya. 2022. № 4. S. 87-105.
20. Kostin A. S. Informatsionno-izmeritel'nye sistemy dlya kontrolya vypolneniya traektorii

aviatsionnoi sistemy // Volnovaya elektronika i infokommunikatsionnye sistemy: materialy XXIV Mezhdunarodnoi nauchnoi konferentsii. Chast' 1. 2021. S. 219-226.

21. Xu Y., Wei Y., Wang D., Jiang K., Deng H. Multi-UAV Path Planning in GPS and Communication Denial Environment // Sensors (Basel). 2023. Vol. 23. № 6. P. 2997.
22. Vera-Yanez D., Pereira A., Rodrigues N., Molina J.P., García A.S., Fernández-Caballero A. Vision-Based Flying Obstacle Detection for Avoiding Midair Collisions: A Systematic Review // J. Imaging. 2023. Vol. 9. P. 194.
23. Alharthi R. Enhancing unmanned aerial vehicle and smart grid communication security using a ConvLSTM model for intrusion detection // Front. Energy Res. 2024. Vol. 12. P. 1491332.

Development of an Application for Design Automation in KOMPAS-3D: Quality Control Table for Base Metal

Savelieva Yulia Olegovna 

PhD in Technical Science

Associate Professor; Branch of SamSTU in Syzran; Samara State Technical University

446001, Russia, Samara region, Syzran, Sovetskaya str., 45, office 404

✉ Savelieva_yu_ol@mail.ru

Il'in Artem Vladimirovich 

Student; Branch of SamSTU in Syzran; Samara State Technical University

446001, Russia, Samara region, Syzran, Sovetskaya str., 45

✉ Savelieva_yu_ol@mail.ru

Luk'yanov Aleksandr Aleksandrovich 

Student; Branch of SamSTU in Syzran; Samara State Technical University

446001, Russia, Samara region, Syzran, Sovetskaya str., 45

✉ Savelieva_yu_ol@mail.ru

Milikhina Kseniya Mihailovna 

Student; Branch of SamSTU in Syzran; Samara State Technical University

446001, Russia, Samara region, Syzran, Sovetskaya str., 45

✉ Savelieva_yu_ol@mail.ru

Abstract. The complexity of compiling quality control tables for base metal TB1 in order to design nuclear power products arises from the necessity to take into account a plenty of requirements from various sources (which may overlap or complement each other): for example, standards such as industry standards (IS/OST), building codes (BC/SNiP), technical specifications, customer requirements, internal rules for their completion, and other regulatory documents. Incorrectly prepared documents submitted to an authorized quality control organization (an entity authorized to conduct conformity assessment in the form of acceptance testing of products intended for use in components of nuclear energy facilities in the Russian Federation) result in financial and time losses for the developer, as well as difficulties in further product acceptance. This paper discloses the development process of a user application (library) that enables the automatic generation and construction of a quality control table for base metal TB1 on the basis of a drawing document in KOMPAS-3D. The application code

(library) for KOMPAS-3D CAD is developed using Delphi, and it implements automated generation of tables TB1 necessary in designing equipment for the nuclear energy sector. One of the key characteristics of a modern engineering simulation system is the ability to integrate automation, for instance, by using subsystems to expand the software package's capabilities, such as user applications (custom programming libraries). This paper formulates the main requirements for the functionality of the user application (library) for KOMPAS-3D. The application code enables reading data from an assembly file opened in KOMPAS-3D, creating a new drawing document, generating a table according to TB1 requirements, and filling it with data from the assembly document.

Keywords: design library creation tools, Delphi, design automation, software library, custom library, TB1, KOMPAS-3D, application, base metal, quality control table

References (transliterated)

1. Norseev S. Razrabotka prilozhenii pod KOMPAS v Delphi, 2013. – 346 s.[Elektronnyi resurs] – URL: <https://norseev.ru/books/kompas-delphi-2013/>
2. Soshkina D.E. Razrabotka pol'zovatel'skoi biblioteki v Kompas-3D / D.E. Soshkina; nauch. ruk. A.A. Dronov // Progressivnye tekhnologii i ekonomika v mashinostroenii : sbornik trudov XIII Vserossiiskoi nauchno-prakticheskoi konferentsii dlya studentov i uchashcheisya molodezhi, 7-9 aprelya 2022 g., Yurga. – Tomsk: Izd-vo TPU, 2022. – S. 34-36.
3. Gushcha A.O. K zadache o razrabotke na C# pol'zovatel'skikh bibliotek standartnykh detalei dlya KOMPAS-3D / A.O. Gushcha, T.D. Stasenکو ; nauch. ruk. V.S. Murashko // Issledovaniya i razrabotki v oblasti mashinostroeniya, energetiki i upravleniya : materialy XXIII Mezhdunar. nauch. – tekhn. konf. studentov, aspirantov i molodykh uchenykh, Gomel', 27–28 apr. 2023 g. : v 2 ch. Ch. 2 / M-vo obrazovaniya Resp. Belarus', Gomel. gos. tekhn. un-t im. P.O. Sukhogo ; pod obshch. red. A.A. Boiko. – Gomel': GGTU im. P. O. Sukhogo, 2023. – C. 268-270.
4. Terekhin R.D., Folomkin A.I. Razrabotka programmnykh bibliotek dlya demonstratsii i postroeniya linii peresecheniya razlichnykh poverkhnostei / Tekhnicheskie nauki – ot teorii k praktike // Sb. st. po materialam LXXI mezhdunar. nauch. – prakt. konf. № 6 (66). Novosibirsk: Izd. ANS «SibAK», 2017. – S. 13-18.
5. Reshetnikova E.S., Savel'eva I.A., Svistunova E.A. Geometricheskoe modelirovanie i razrabotka pol'zovatel'skikh bibliotek pri proektirovanii ob'ektov mashinostroeniya // Programmnye sistemy i vychislitel'nye metody. 2020. № 1. S. 1-7. DOI: 10.7256/2454-0714.2020.1.32292 URL: https://nbpublish.com/library_read_article.php?id=32292
6. Kidruk M.I. Konstruktorskie biblioteki i instrumenty dlya ikh sozdaniya v sisteme KOMPAS-3D. Chast' 1. V pomoshch' konstruktoru – biblioteki, raschetnye moduli i spravochniki KOMPAS-3D // stat'ya v zhurnale «SAPR i grafika». Mashinostroenie, № 1, 2006 [Elektronnyi resurs] – URL: <https://sapr.ru/article/14850>
7. Kidruk M.I. Konstruktorskie biblioteki i instrumenty dlya ikh sozdaniya v sisteme KOMPAS-3D. Chast' 2. Sdelai sam, ili Kak sozdat' svoju biblioteku dlya KOMPAS-3D // stat'ya v zhurnale «SAPR i grafika». Mashinostroenie, № 2, 2006 [Elektronnyi resurs] – URL: <https://sapr.ru/article/15598>
8. Forum pol'zovatelei PO Askon. [Elektronnyi resurs] – URL: <http://forum.ascon.ru/index.php>
9. OST 108.004.10-86. Otrasleyvoi standart. Programma kontrolya kachestva izdelii atomnoi energetiki. – Deistvuet s 01.01.2007 – M.: Izmenenie № 12, 2019. 143 s. 1.

Norseev S. Razrabotka prilozhenii pod KOMPAS v Delphi, 2013. – 346 s. [Elektronnyi resurs] – URL: <https://norseev.ru/books/kompas-delphi-2013>

10. RD EO 1.1.2.01.0713-2019. Otsenka sootvetstviya v formakh priemki, ispytaniy produktov dlya atomnykh stantsii. Polozhenie. – Vzamen RD EO 1.1.2.01.0713-2013 «Polozhenie ob otsenke sootvetstviya v forme priemki i ispytaniy produktov dlya atomnykh stantsii». – Vved. s 20.02.2020 – M.: Aktualizatsiya 01.01.2021. – 60 s.
11. Forum DWG.RU dlya obsuzhdeniya problem proektirovaniya, konstruirovaniya v oblasti stroitel'stva, arkhitektury, mashinostroeniya, smezhnykh otraslei. [Elektronnyi resurs] – URL: <https://forum.dwg.ru/showthread.php?t=167100>

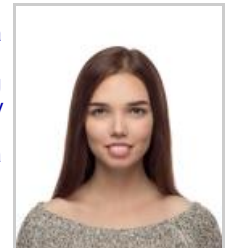
Development of a methodology for testing a mobile application based on eye-tracking technology

Pletneva Alisa Olegovna

Graduate student, Department of Higher School of Design and Architecture, Peter the Great St. Petersburg Polytechnic University

Office 29, Politechnicheskaya str., St. Petersburg, 194064, Russia

✉ pletneva.alisa2010@yandex.ru



Yanchus Viktor Edmundasovich

PhD in Technical Science

Yanchus Viktor Edmundasovich - Candidate of Technical Sciences, lecturer, Higher School of Design and Architecture, Peter the Great St. Petersburg Polytechnic University

194064, Russia, Saint Petersburg, Politechnicheskaya str., 29, office 29AF

✉ victorimop@mail.ru



Abstract. A mobile phone is used every day for various tasks, for example, for mobile communication, for communicating on social networks, for obtaining information from the Internet, as well as for mobile applications. A smartphone is a small device that contains a lot of information that needs to be processed and analyzed in order for the user to achieve the goal. From a design point of view, to evaluate a mobile application, this work identified 4 factors influencing the perception of information in a mobile application: color scheme, compositional solution, font and the number of letters in control words. An experiment was developed that was carried out using eye-tracking technology in a special software module. It includes 72 stimuli and 1 set task that participants in the experiment must solve. The experiment demonstrated that its participants solved the problem fastest in blue-orange and white-black color combinations of stimulus material, and also the fastest in solving problems with large elements of the compositional solution. An interesting observation is that long words were perceived by the subjects faster than short ones. This study showed that the developed methodology can be used for testing mobile applications. The testing was successful, recommendations were identified that can help in developing a mobile application: light and dark themes in which different fonts will be displayed: light theme – sans-serif font, dark theme – serif font; mobile application elements in black and white and blue and orange color combinations; long words in mobile application elements and large square tiles located on the screens of the mobile application. The technique showed results that require additional, more in-depth research.

Keywords: Color solution, Peripheral vision, Central vision, Saccade, Fixations, Eye-tracking

technology, Compositional solution, Mobile application, Font, Control words

References (transliterated)

1. Moreva D.V., Kamal'dinova Z.F. Osobennosti testirovaniya mobil'nykh prilozhenii // Tsifrovye tekhnologii: nastoyashchee i budushchee. 2022. S. 87-92.
2. Zayats A.M., Vasil'ev N.P. Vvedenie v gibridnye tekhnologii razrabotki mobil'nykh prilozhenii. Spb.: «Lan'», 2022.
3. Sugaipov S.A.A., Gerikhanov Z.A. Iskustvo razrabotki mobil'nykh prilozhenii // Tendentsii razvitiya nauki i obrazovaniya. 2023. № 98-10. S. 134-136.
4. Pantileikin N.V. Mobil'nye prilozheniya i ikh vidy // Nauchno-metodicheskii elektronnyi zhurnal «Kontsept». 2016. T. 26. S. 776-780.
5. Bogdanova V.S. Printsipy razrabotki interfeisa mobil'nogo prilozheniya // Nauka i proizvodstvo Urala. 2021. T. 17. S. 108-110.
6. Pletneva A.O., Yanchus V.E. Razrabotka metodiki testirovaniya mobil'nogo prilozheniya na osnove tekhnologii ai-trekinga // Trudy mezhdunarodnoi konferentsii po komp'yuternoi grafike i zreniyu «Grafikon». 2023. № 33. S. 959-968.
7. Gimatdinov R.S. Tendentsii razvitiya sfery razrabotki mobil'nykh prilozhenii // Voprosy studencheskoi nauki. 2019. № 6(34). S. 359-362.
8. Ovchinnikov M.A., Ovchinnikova M.A. Otsenka kachestva pol'zovatel'skogo interfeisa pri pomoshchi tekhnologii otslezhivaniya vzglyada // Sovremennye informatsionnye tekhnologii v obrazovanii, nauke i promyshlennosti. 2023. S. 75-77.
9. Laptev M.V., Yanchus V.E., Laptev V.V. Vyyavlenie zon interesa pol'zovatelya pri vizualizatsii dannykh s ispol'zovaniem metoda ai-trekinga // ООО «PSK "FRAM"». 2023. S. 49-62.
10. Kheifits A.E., Yanchus V.E., Borevich E.V. Metodika provedeniya eksperimental'nogo issledovaniya vospriyatiya graficheskogo interfeisa s primeneniem tekhnologii ai-trekinga // Programmnye sistemy i vychislitel'nye metody. 2022. № 2. S. 52-62.
11. Sarycheva Yu.Yu., Fedorov V.O. Primenenie ustroystv otslezhivaniya vzglyada pri issledovanii interfeisov programmnykh produktov // Nauchno-prakticheskii elektronnyi zhurnal. 2022. T. 12. № 8. S. 52-63.
12. Samogorov V.A., Konkina E.D. Teoriya tsvetovykh kontrastov Iokhannesa Ittena // Samarskii vestnik. 2021. T. 11. № 3. S. 97-103.
13. Yanchus V.E., Borevich E.V. Issledovanie znacheniya tsvetovogo resheniya v protsesse garmonizatsii kinokadra // Nauchno-tekhnicheskie vedomosti SPbGPU. 2016. №4. S. 53-68.
14. Orlov P.A., Laptev V.V., Ivanov V.M. K voprosu o primenении sistem ai-trekinga // Nauchno-tekhnicheskie vedomosti SPbGPU. 2014. №5(205). S. 84-94.
15. Sheffe G. Dispersionnyi analiz. Moskva: Nauka, 1980.
16. Borovikov V.P. Statistika: iskustvo analiza dannykh na komp'yutere. Spb: Piter, 2003.