

Разработка программно-аппаратного комплекса для управления группой беспилотных транспортных средств на основе робастных алгоритмов компьютерного зрения

А. М. Белов¹, П. Ю. Беляев¹, И. И. Виксин¹, Ю. В. Ким¹,
В. С. Радабольский^{1✉}, Т. А. Турушев¹, С. С. Чупров^{1,2}

¹ Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина), Санкт-Петербург, Россия

² Колледж вычислительной техники и информационных наук Голисано Рочестерского технологического института, Рочестер, США

✉ radabolsky2017@gmail.com

Аннотация. Повсеместное введение в эксплуатацию автоматизированных робототехнических систем требует гарантий эффективности и безопасности их использования. Для проверки этих критериев разрабатываются макеты, имитирующие внешнюю среду с похожими на среду эксплуатации параметрами. Особенность построения систем, ориентированных на выполнение задач в реальных условиях, состоит в их устойчивости к изменениям внешней обстановки. Цель исследования заключается в разработке такого программно-аппаратного комплекса для управления группой беспилотных транспортных средств (БТС), который был бы устойчив к условиям агрессивного влияния внешних факторов. В рамках исследования был разработан прототип БТС и упрощенная версия среды, в которой планируется его использовать. В основе центра принятия решений каждого БТС лежит система компьютерного зрения, включающая в себя алгоритмы позиционирования, коррекции прямолинейного движения, а также детектирования других транспортных средств. Для верификации разработанной системы были проведены экспериментальные исследования с целью определения влияния таких внешних факторов, как изменение уровня и тона освещения, а также изменение внешнего вида поверхности, предназначенной для передвижения. Результаты исследования показали, что разработанная и реализованная система компьютерного зрения БТС отвечает требованиям надежности и устойчивости независимо от условий внешней среды и может гарантировать: исключение нарушений запланированного маршрута, точность локализации в пределах 10 % погрешности, точность детектирования других БТС с вероятностью 84 %, а также возможность автономной работы без нарушений в течение двух часов. Разработанный программно-аппаратный комплекс с макетом внешней среды может быть использован для исследования и тестирования алгоритмов безопасности и управления «умным городом», а также для других задач, требующих предварительного прототипирования.

Ключевые слова: беспилотные транспортные средства, компьютерное зрение, нейронные сети, предварительная обработка изображений, робототехнические системы

Для цитирования: Разработка программно-аппаратного комплекса для управления группой беспилотных транспортных средств на основе робастных алгоритмов компьютерного зрения / А. М. Белов, П. Ю. Беляев, И. И. Виксин, Ю. В. Ким, В. С. Радабольский, Т. А. Турушев, С. С. Чупров // Изв. СПбГЭТУ «ЛЭТИ». 2023. Т. 16, № 6. С. 52–69. doi: 10.32603/2071-8985-2023-16-6-52-69.

Финансирование. Работа выполнена при поддержке Министерства науки и высшего образования РФ «Госзадание» № 075-01024-21-02 от 29.09.2021 (проект FSEE-2021-0014).

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Software and Hardware Bundle for Controlling a Group of Unmanned Vehicles Based on Robust Computer Vision Algorithms

A. M. Belov¹, P. Yu. Belyaev¹, I. I. Viksnin¹, Yu. V. Kim¹,
V. S. Radabolsky^{1✉}, T. A. Turushev¹, S. S. Chuprov^{1,2}

¹ Saint Petersburg Electrotechnical University, Saint Petersburg, Russia

² Golisano College of Computing and Information Sciences,
Rochester Institute of Technology, Rochester, USA

✉ radabolsky2017@gmail.com

Abstract. The widespread adoption of automated robotic systems requires their confirmed efficiency and safety. To test these criteria, experimental testbeds that simulate an external environment under the conditions similar to the operating environment are developed. Systems focused on performing tasks under real conditions should be robust against the dynamic external environment. In this work, we develop a combination of software and hardware components for controlling a group of unmanned vehicles (UVs) based on computer vision algorithms that would be robust to external environmental factors. As part of our investigation, UV prototypes and their testing environment are developed. The action decision-making of each UV is based on a computer vision system, which incorporates algorithms for positioning, movement correction, other vehicle and obstacle detection. To verify the developed system, we conduct an empirical study into how the UV decision-making is affected by such external environmental factors, as changes in the level and tone of lighting, as well as the appearance of the surface. The experimental results show that the developed and implemented UV computer vision system meets the established robustness and stability requirements regardless of the tested environmental conditions. The developed system minimizes deviations from the planned route, reduces the localization error to 10 % at most, provides the accuracy of 84 % of other UVs detection, and maintains UVs operation without failures for two hours. The developed hardware and software bundle and testbed can be used for security and smart city control algorithms verification, as well as for other tasks requiring UVs prototyping.

Keywords: unmanned vehicles, computer vision, neural networks, image preprocessing, robotic systems

For citation: Software and Hardware Bundle for Controlling a Group of Unmanned Vehicles Based on Robust Computer Vision Algorithms / A. M. Belov, P. Y. Belyaev, I. I. Viksnin, Yu. V. Kim, V. S. Radabolsky, T. A. Turushev, S. S. Chuprov // LETI Transactions on Electrical Engineering & Computer Science. 2023. Vol. 16, no. 6. P. 52–69. doi: 10.32603/2071-8985-2023-16-6-52-69.

Funding. The work was carried out with the support of the Ministry of Science and Higher Education of the Russian Federation «State Task» № 075-01024-21-02 dated 29.09.2021 (project FSEE-2021-0014).

Conflict of interest. The authors declare no conflicts of interest.

Введение. В настоящее время использование и развитие информационных технологий для автоматизации рутинных и трудоемких процессов – это перспективное направление научных исследований. Интеграция автономных робототехнических систем в различные сферы деятельности человека позволяет повысить эффективность выполнения задач за счет снижения участия человека в их выполнении [1], [2]. В данном исследовании будут рассмотрены наземные беспилотные транспортные средства (БТС) как наиболее распространенные автономные робототехнические системы. БТС используются для решения многих

задач, в частности для мониторинга местности и транспортировки грузов.

Для извлечения состояния окружающей среды и принятия решения относительно дальнейших действий БТС использование графической информации, полученной с бортовой камеры, позволит оперативно принимать решения без участия других датчиков. На основе компьютерного зрения была построена автономная модульная система управления БТС. В рамках данного исследования рассматривается система компьютерного зрения, в состав которой входят модули:

- коррекции прямолинейного движения;
- позиционирования;
- детектирования других транспортных средств.

Разработанная система призвана работать в условиях потенциальных искажений графической информации. Кроме того, для корректности обработка изображений должна происходить в реальном времени с учетом ограниченной производительности вычислительной системы. Для борьбы с искажениями используются различные методы предобработки: размытие по Гауссу, операции математической морфологии, адаптивное выравнивание гистограммы, алгоритм поиска границ и т. д. Для извлечения информации более сложной структуры, например для определения местонахождения другого транспортного средства на изображении с камеры, используется сверточная нейронная сеть [3]. Помимо получения графической информации для обеспечения корректного взаимодействия, БТС также обмениваются между собой информацией о местонахождении друг друга.

Постановка задачи. К основным параметрам при разработке систем управления БТС относятся их автономность и низкий уровень вмешательства человека [4]. Автономность системы подразумевает ее своевременное и правильное реагирование на изменения внешней среды. Таким образом, для обеспечения автономности необходимо, чтобы каждый элемент системы обладал наиболее полной информацией о внешней среде и событиях, которые происходят в ней. Под событием подразумевается изменение внешних условий около каждого БТС: смена координат, изменение направления движения, проезд других транспортных средств. Для каждого события должна быть предусмотрена реакция, обеспечивающая дальнейшее автономное функционирование системы. Координаты должны меняться своевременно и отражать положение БТС в настоящий момент времени – изменение направления движения не должно повлечь за собой выезда за пределы проложенного маршрута, встреча с другим БТС не должна закончиться столкновением с ним. Наиболее подходящим датчиком для оперативного и полного сбора информации подобного рода служит камера, так как из графического представления можно извлечь наибольшее количество признаков.

Таким образом, цель данной работы заключается в разработке системы компьютерного зрения

для организации движения БТС в условиях структурной среды с заранее известным планом местности с учетом изменяющихся условий видимости.

Описание разрабатываемых элементов. Для разработки и тестирования системы управления БТС был спроектирован и собран имитационный стенд, в состав которого входят модель внешней среды и прототипы БТС (3 шт.).

Модель внешней среды представлена в виде клеточного поля площадью 3 м², состоящего из набора черных и белых клеток с искусственными препятствиями в виде выделяющихся прямоугольных областей. Количество клеток, доступных для проезда БТС, – 65. Модель внешней среды представлена на рис. 1.

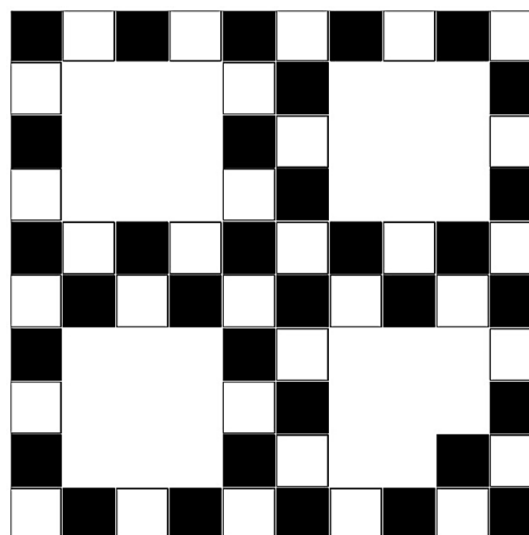


Рис. 1. Модель внешней среды
Fig. 1. Testbed model

При проектировании модели внешней среды стояла задача выбрать такую, которая удовлетворяла бы критериям масштабируемости и устойчивости. Всем этим критериям удовлетворяет структурная среда. Для имитации структурной среды была выбрана модель клеточного поля, так как клетка – это простейшая структурная единица, которую легче всего воспроизвести.

Стенд предназначен для проверки работы системы компьютерного зрения для наземного БТС, поэтому модель объекта управления должна удовлетворять следующим критериям:

- автономность. Объект должен самостоятельно, без участия оператора, обрабатывать информацию, полученную от внешней среды, и принимать решения о дальнейших передвижениях;
- маневренность. Объект должен иметь возможность перемещаться в любом направлении;

– компактность. Объект должен иметь небольшие размеры;

– коммуникативность. Объект должен иметь возможность связываться с другими объектами.

Для имитации взаимодействия между несколькими БТС было создано 3 прототипа, удовлетворяющих вышеперечисленным свойствам. Прототип БТС состоит из аппаратной и программной частей. Аппаратная часть включает в себя комплектующие, необходимые для соблюдения установленных условий. Программная часть состоит из нескольких взаимосвязанных модулей. Основным модулем принятия решения является система компьютерного зрения.

Для выполнения требования компактности состав вычислительной системы включает в себя минимально необходимые части: одноплатный бортовой компьютер для управления и анализа системы непосредственно самим БТС (автономность), периферийные платы управления – устройства, связывающие центр принятия решения с необходимыми частями системы. Общая схема аппаратного обеспечения БТС представлена на рис. 2, а состав аппаратной части – в табл. 1.

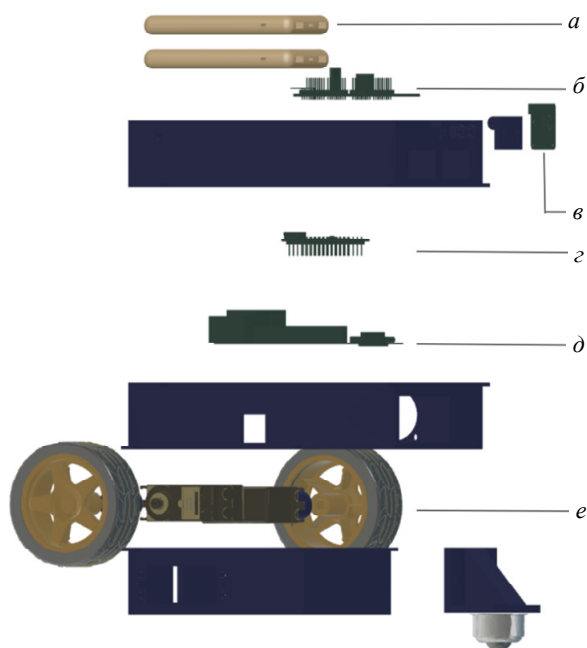


Рис. 2. Основные элементы аппаратного обеспечения: *a* – раздельное питание для моторов и вычислительной системы; *b* – ШИМ-контроллер для управления моторами; *в* – камера Raspicam V2; *г* – микроконтроллер Arduino Nano; *д* – главный бортовой компьютер Rpi 4 Model B; *е* – электромоторы на базе MG996R

Fig. 2. Main hardware components. *a* – separated power for motors and computational units; *b* – PWM controller for motor control; *в* – raspicam V2 camera; *г* – micro controller Arduino Nano; *д* – main computer Rpi 4 Model B; *е* – electric motors based on MG996R

Табл. 1. Аппаратный состав БТС
Tab. 1. Hardware composition of the UV

Модуль	Состав
Принятие решений	Raspberry Pi 4
Датчики	Raspberry Pi Camera Module 2
Движение	Arduino Nano Контроллер PCA9685 с настраиваемой частотой ШИМ Сервомоторы MG996R (2 шт.)
Питание	Внешний аккумулятор TFN. Внешний аккумулятор Buro T4

Выполнение условия маневренности БТС обеспечивается двумя модифицированными сервомоторами MG996R и микроконтроллером управления.

Сервомоторы MG996R используют потенциометр для определения угла, на который повернут выходной вал. Для вращения на определенный угол MG996R используют систему обратной связи, которая увеличивает или уменьшает напряжение на моторе в зависимости от текущего угла сервопривода и установленного угла. В процессе модификации потенциометр отсоединяется от выходного вала, а его ручка фиксируется в положении с отклонением 0° , а также удаляется защитный зубчик на шестерне выходного вала. В результате получается недорогой мотор с высоким крутящим моментом. При этом можно использовать обратную связь для контроля скорости вращения: чем больше угол подачи на контроллер сервомотора, тем выше скорость вращения.

Роль управляющего микроконтроллера выполняет Arduino Nano. Так как сервомоторы требуют напряжения 5 В, а максимальное значение напряжения в Arduino Nano 3 В, то для управления необходимо дополнительное устройство. Подключение к сервомоторам выполняется через плату PCA9685.

Для питания системы используется 2 внешних аккумулятора: для вычислительной системы и сервомоторов. Необходимость в дополнительном аккумуляторе связана с высоким пусковым током сервомоторов, который не может обеспечить один аккумулятор при подключении к нему нескольких устройств. Второй аккумулятор используется для питания Raspberry Pi, которая в свою очередь питает камеру через шлейф, а также микроконтроллер – через USB. Для предотвращения автоматического отключения внешнего аккумулятора при долгом отсутствии потребления напряжения к данному аккумулятору подключена дополнительная нагрузка в виде светодиода.

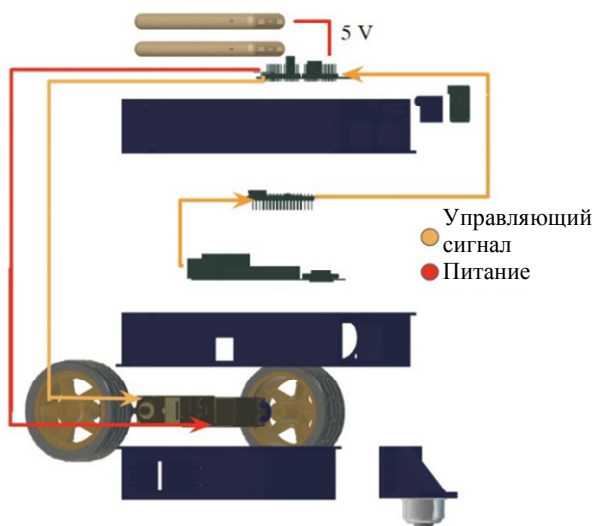


Рис. 3. Схема питания БТС
 Fig. 3. Power scheme of the UV

Схема питания и связи между аппаратными составляющими системы изображена на рис. 3.

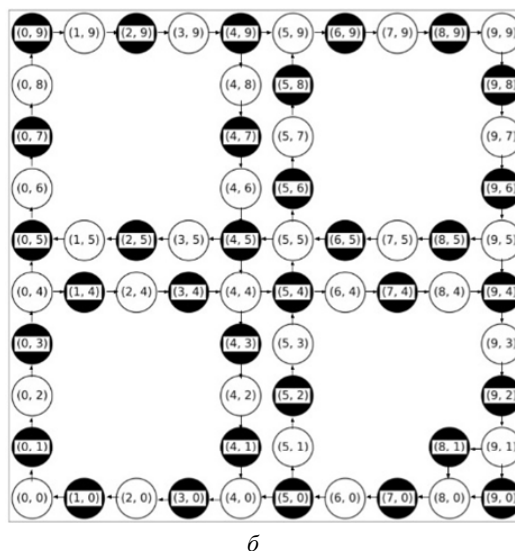
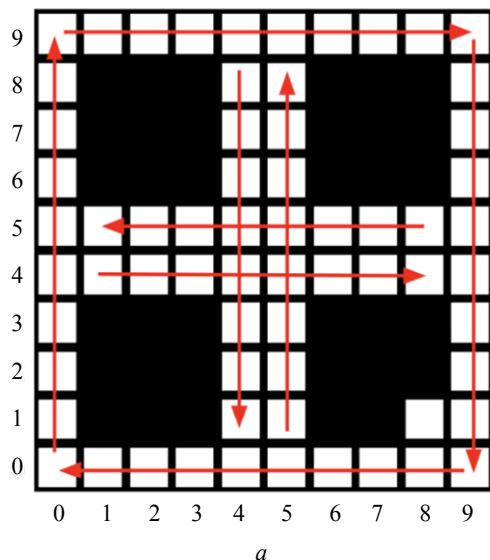


Рис. 4. Представление стенда (а) в виде графа (б)
 Fig. 4. Stand (a) graph representation (б)

Программное обеспечение БТС. Для навигации по клеточному полю и эффективному построению маршрутов используется представление поля стенда в виде связного ориентированного графа, направления ребер которого совпадают с разрешенным направлением движения. Схема представления изображена на рис. 4.

Маршрут между любыми двумя точками строится по алгоритму поиска кратчайшего пути Дейкстры [5].

Коммуникация БТС обеспечивается модулем связи, основанным на общей сети Wi-Fi. Посредством Wi-Fi-роутера или компьютера с Wi-Fi мо-

дулем создается сеть WLAN, к которой подключаются БТС. Схема связи представлена на рис. 5.

Идентификация БТС происходит по выданным в этой сети IP-адресам. Для лучшей надежности используется протокол TCP. Так как Wi-Fi-модуль находится на плате Raspberry Pi, БТС может сразу же получать сообщения и использовать в программных модулях. Поскольку БТС функционируют в группе, помимо связи и логики построения маршрута нужна система, способная анализировать окружающую среду и помогать принимать решения на основе результатов анализа. Для решения этой задачи была разработана система компьютерного зрения.

На вход системе подается непрерывный поток изображений с камеры, результатом анализа которого являются:

- корректировка движения БТС для выполнения условия прямолинейного движения;

- определение местоположения БТС в системе координат на основе визуальной одометрии;
- распознавание других БТС.

Проблемы, которые могут возникнуть при обработке графической информацией системой компьютерного зрения:

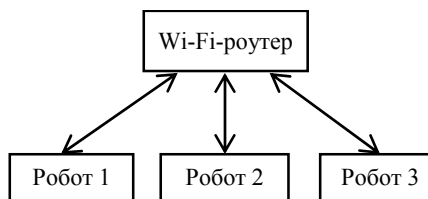


Рис. 5. Схема связи
 Fig. 5. Communication scheme

– изменение цветовой контрастности при различном уровне освещения;
 – возникновение шума от камеры;
 – потеря информации при сжатии изображений;
 – возникновение шума от внешней среды: наличие отражений, бликов и теней, загрязненность и замятие полотна.

Корректировка движения. Модуль корректировки движения предназначен для исключения выезда БТС за пределы поля и гарантии прямолинейного движения. Данный модуль состоит из двух частей:

– аппаратная, представленная в виде двух сервомоторов, управляемых по напряжению;
 – программная, принимающая решение на основе потокового изображения с камеры.

Алгоритм коррекции предполагает, что БТС всегда должен стремиться изменить направление своего движения в сторону центра той клетки, на которую направлена камера в текущий момент времени. Определение центра клетки состоит из двух этапов:

- 1) обнаружение контуров клетки, стоящей впереди;
- 2) определение центраоида клетки как центра обнаруженного контура.

Таким образом, основное узкое место алгоритма – это поиск правильного контура. На алгоритм поиска может повлиять множество внешних условий:

- 1) помехи камеры;
- 2) изменение цветовой интенсивности при изменении условий освещения;
- 3) загрязненность и замятие полотна стенда, наличие бликов и теней.

При применении классических методов поиска контуров без предобработки возможно возникновение ошибок. Например, при изменении уровня освещенности границы между клетками могут размываться, в результате чего контуры

нескольких клеток могут сливаться в единый контур. Изменение поверхности полотна, например появление на нем складок или грязи, влечет за собой неправильное выделение контура. Пример подобной ситуации изображен на рис. 6.

Таким образом, для правильной коррекции БТС необходимо применение комплекса средств предварительной обработки исходного изображения.

Рассмотрим общий алгоритм коррекции БТС с учетом предварительной обработки. На вход алгоритма подается изображение с видеопотока камеры БТС, на котором выбирается точка интереса – клетка, по которой будет происходить выравнивание. К изображению применяется метод Canny [6], с помощью которого находят границы перепада интенсивности на изображении, после чего к полученному бинарному изображению применяется обработка методами математической морфологии.

Для определения отклонения БТС от прямолинейной траектории можно использовать знания о структурных свойствах среды. Так как заранее известно, что границы поля четко обозначены, текущее направление движения БТС определяется сравнением координат центрального пикселя изображения и центра клетки поля. Общий алгоритм представлен на рис. 7.

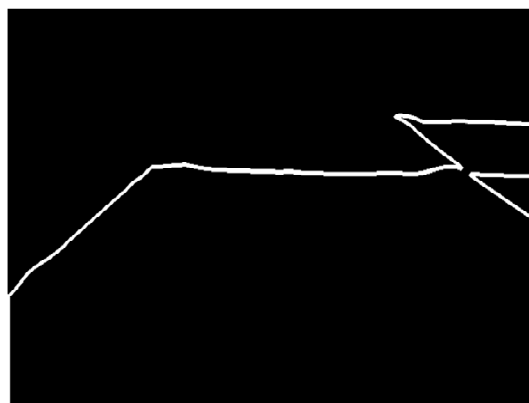
Определение центра клетки начинается с поиска границ структурных элементов (клеток поля) и определения интересующей клетки. Для нахождения границ используется алгоритм Canny.

Для усиления выделенных и фильтрации ненужных границ используются операции математической морфологии.

Дилатация применяется для расширения границ, выделенных Canny. Результат рассчитывается по формуле



а



б

Рис. 6. Пример неправильного выделения контура: а – исходное изображение; б – выделенный контур
 Fig. 6. Example of incorrect boundaries selection: a – original image; б – selected contour

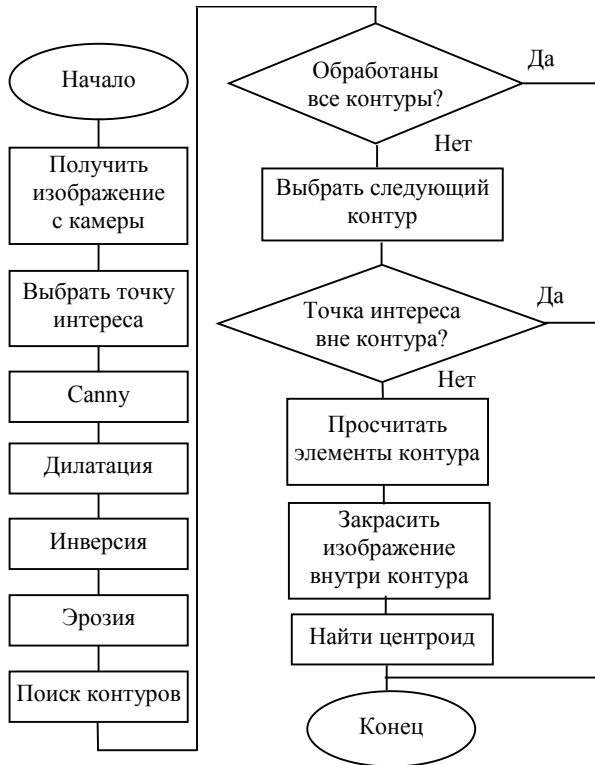


Рис. 7. Общий алгоритм определения отклонения БТС
Fig. 7. General algorithm for determining the UV's deviation from a planned movement trajectory

$$A \oplus B = \bigcup_{b \in B} A_b,$$

где A – бинаризованная картинка; B – структурный элемент; A_b – смещение A на b .

Эрозия позволяет увеличить расстояния между выделенными клетками:

$$A \ominus B = \bigcap_{b \in B} A_{-b},$$

где A_{-b} – смещение A на $-b$.

Пример применения алгоритма поиска контура клетки представлен на рис. 8.

Для поиска центра клетки выбирается контур, который содержит пиксель с координатами центра изображения. Затем находится центр тяжести этой фигуры с помощью поиска моментов изображения [7] по формулам

$$M_{i,j} = \sum_x \sum_y x^i y^j I(x,y),$$

$$\text{Centroid} : \{\bar{x}, \bar{y}\} = \left\{ \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right\}.$$

Алгоритм поиска центра тяжести приведен на рис. 9.

Примененные методы предварительной обработки позволили решить проблему слияния нескольких контуров в один. Сравнение результатов работы алгоритма без применения обработки и с использованием ее представлены на рис. 10.

Полученное от модуля коррекции отклонение центра кадра затем используется для задания мощности P сервоприводов, находящейся в границах от P_{\min} до P_{\max} .

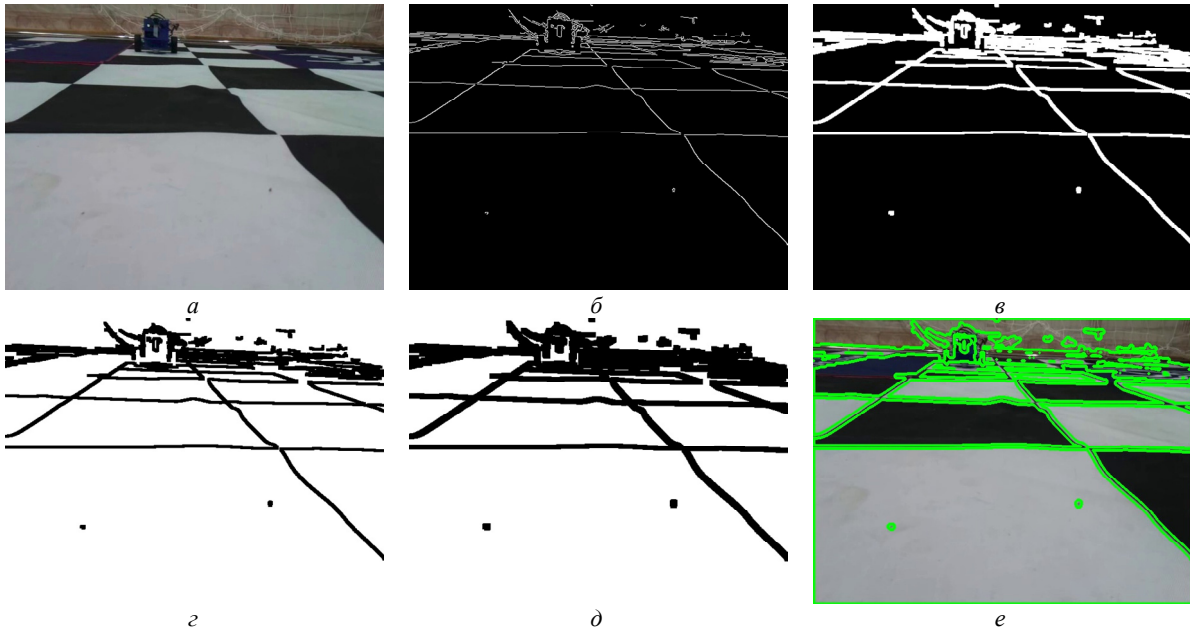


Рис. 8. Пример применения алгоритма поиска контуров клетки: a – исходное изображение; b – порог Canny 100, 200; $в$ – дилатация ядром 5×5 ; $г$ – инверсия; $д$ – эрозия ядром 5×5 ; e – поиск контуров
Fig. 8. Results from the algorithm for searching the boundaries of a square: a – original image; b – Canny threshold 100, 200; $в$ – dilation with 5×5 core; $г$ – inversion; $д$ – erosion with 5×5 core; e – finding boundaries

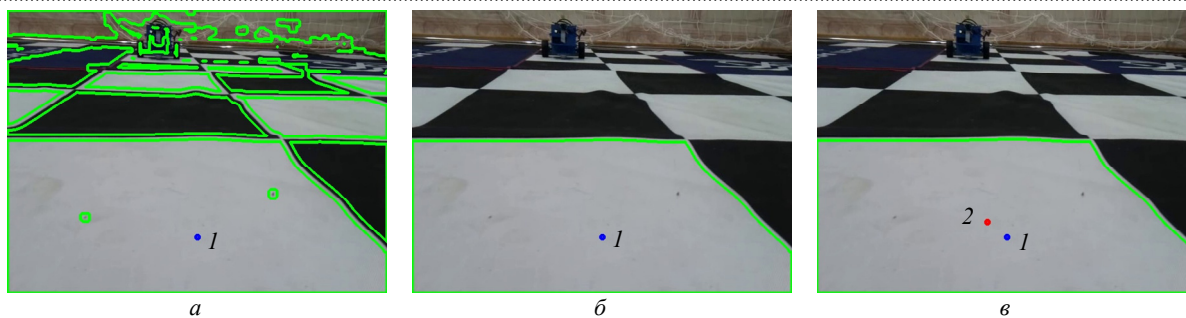


Рис. 9. Пример работы алгоритма поиска центра контура: а – исходное изображение с контурами; б – поиск контура, содержащего точку интереса; в – поиск центра контура; точка 1 – точка интереса; точка 2 – центр клетки containing the interest point; в – finding the centroid; 1 (blue point) – interest point; 2 (red point) – center of the square

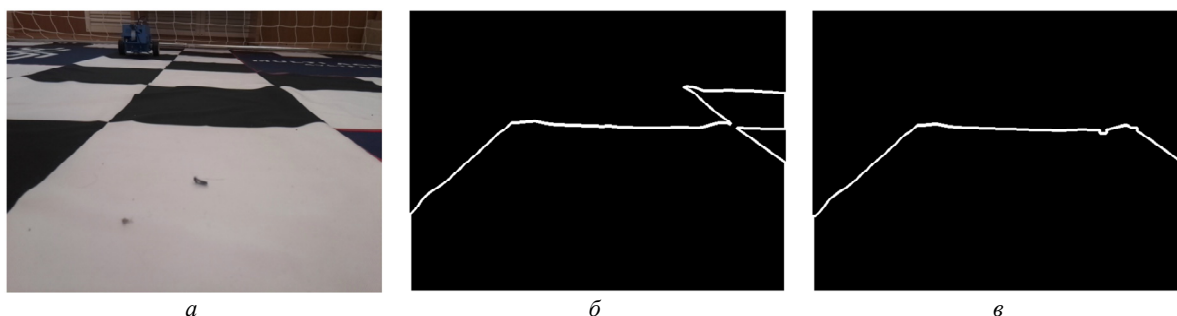


Рис. 10. Сравнение результатов алгоритма поиска контуров: а – оригинальное изображение; б – поиск контура без предобработки; в – поиск контура с предобработкой
Fig. 10. Comparison between the results obtained by the algorithm for finding contour: а – original image; б – contour search without preprocessing; в – contour search with preprocessing

Ширина кадра, полученного с видеокамеры, задается в пикселях и обозначается w . Отклонение d , полученное от модуля коррекции, также измеряется в пикселях, причем $|d| \in [0, w/2]$, так как отклонение рассчитывается относительно центра кадра. На рис. 11 представлено схематичное изображение заданных величин.

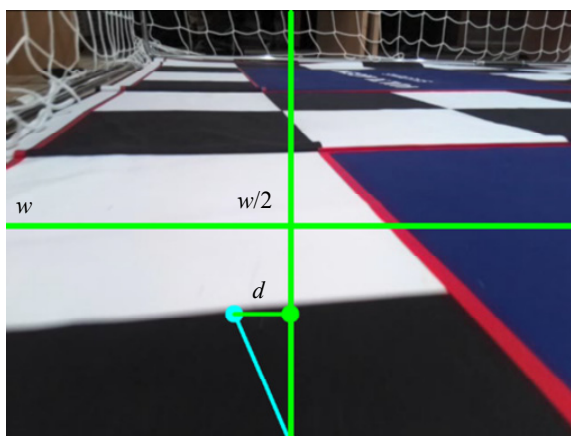


Рис. 11. Схематичное изображение заданных величин
Fig. 11. Representation of used values

Отклонение d нормализуется по его максимальным и минимальным значениям:

$$d' = \frac{d - \min(d)}{\max(d) - \min(d)}$$

С помощью полученных данных можно управлять движением сервоприводов, построив модель изменения мощностей на каждом из них. Главное требование для модели – сильная реакция на изменение отклонения d , чтобы БТС успел изменить мощность на сервоприводах и вернуться на курс движения. По результатам в качестве функции с высокой скоростью роста экспериментов была выбрана параметрическая модель изменения мощности сервопривода $P(d')$, в которой в качестве параметра выступает скорость возрастания мощности a :

$$P(d') = (1 - a)^{d'}, \quad a \in [0; 1].$$

Для масштабирования нормализованных значений мощности к значениям, используемым сервоприводом i , используется формула

$$P_{i \text{ scale}} = \frac{P_i - \min(P_i)}{\max(P_i) - \min(P_i)} (P_{\max} - P_{\min}) + P_{\min}$$

Позиционирование. Для навигации по стенду БТС использует информацию о цвете клетки, расположенной перед ним, в комбинации с внутренней одометрией. Координатное пространство стенда построено в форме взаимно однозначного соответствия между клетками поля и двумерными осями координат. Точкой начала координат счита

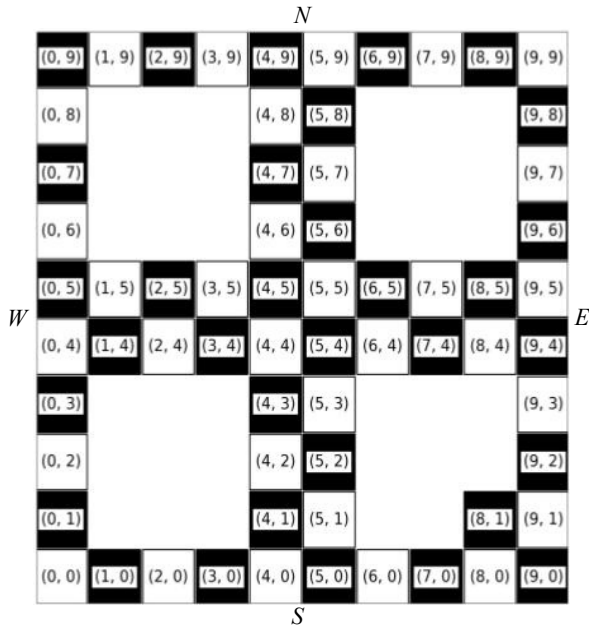


Рис. 12. Координатное поле стенда
 Fig. 12. Coordinate markup of the testbed

ется левый нижний угол поля. Соответствие между координатами и полем показано на рис. 12.

Координаты БТС изменяются при регистрации им изменения внешней среды (в случае стенда – изменение цвета клетки, находящейся перед БТС). Направление изменения задается в соответствии с хранимым внутренним состоянием БТС. Состояния обозначены латинскими буквами *W*, *S*, *E*, *N* и характеризуют направление вида камеры в данный момент времени. При поворотах внутреннее состояние системы меняется в зависимости от первоначального состояния и направления

поворота. Общий алгоритм изменения координат представлен на рис. 13.

Для определения доминантного цвета клетки используется алгоритм, преобразующий цветное изображение в черно-белое, после чего происходит удаление шума и бинаризация изображения с подсчетом количества светлых и темных пикселей. Исходя из этого соотношения происходит определение, изменился цвет клетки или нет. Наличие этапа предобработки изображения вызвано изменениями условий видимости и освещенности, а также шумов различной природы. Самая частая помеха в работе модуля – загрязненность поля, из-за которой может быть неправильно определен текущий цвет клетки при определении цвета по одному пикселю. Пример такой ошибки приведен на рис. 14, текстом обозначен цвет клетки, на которой изображен черный круг.

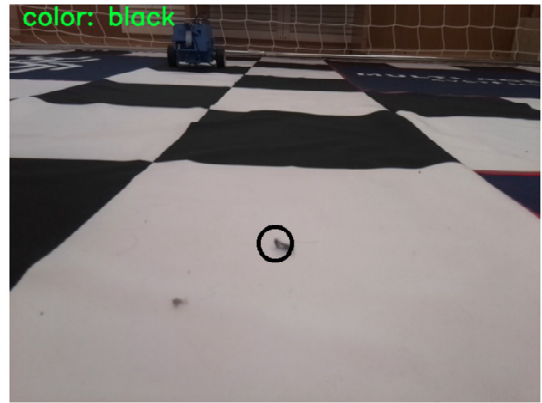


Рис. 14. Пример ошибки модуля позиционирования
 Fig. 14. Example of the localization module error

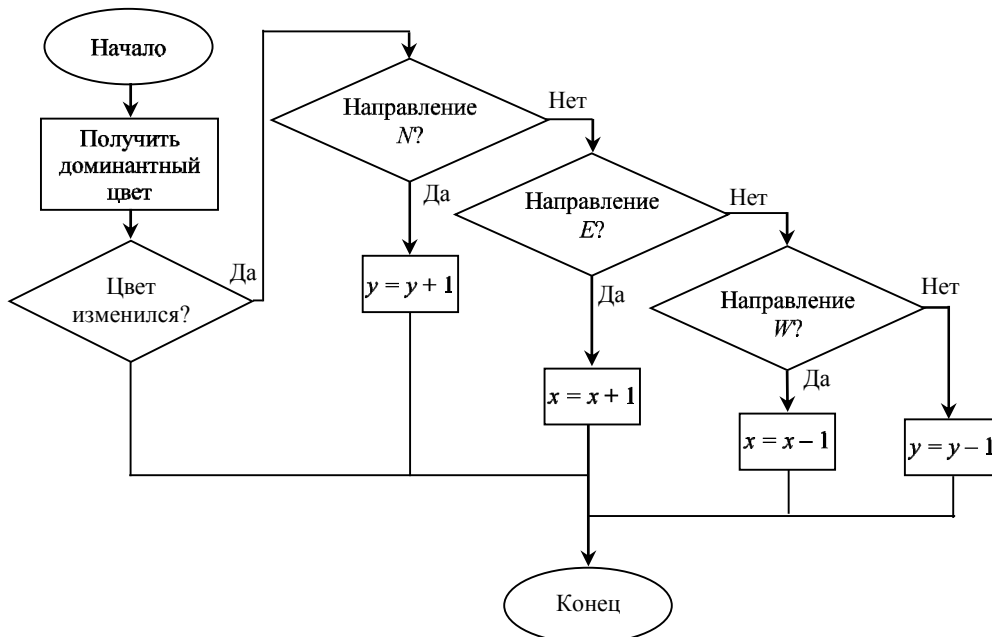


Рис. 13. Общий алгоритм изменения координат
 Fig. 13. General algorithm for a coordinate change

Общий алгоритм решения обозначенной проблемы представлен на рис. 15.



Рис. 15. Общий алгоритм определения цвета клетки
 Fig. 15. General algorithm for determining the square color

Для удаления шумов с изображения используется размытие по Гауссу, определяемое согласно выражению

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-x^2/(2\pi^2)}$$

В исследовании использовалась константа $\sigma = 10$.

Затем к полутоновому изображению, обработанному с помощью фильтра Гаусса, применяется бинаризация методом Оцу [8]. Бинаризация необходима для определения преобладания темных или светлых цветов. Метод Оцу разделяет полутоновое изображение на два кластера с порогом, определяемым минимизацией взвешенной дисперсии:

$$\sigma^2(t) = w_1(t)w_2(t)[\mu_1(t) - \mu_2(t)]^2$$

Для предотвращения влияния шумовых пикселей цвет вычисляется по количеству пикселей в окне интереса с использованием формулы

$$\text{color}(X, Y, \text{offset}) = \sum_{y=Y-\text{offset}}^{Y+\text{offset}} \sum_{x=X-\text{offset}}^{X+\text{offset}} G(x, y)$$

Если количество пикселей со значением 255 преобладает над значением 0, то перед БТС находится клетка белого цвета, иначе – черного.

Визуализация алгоритма приведена на рис. 16.

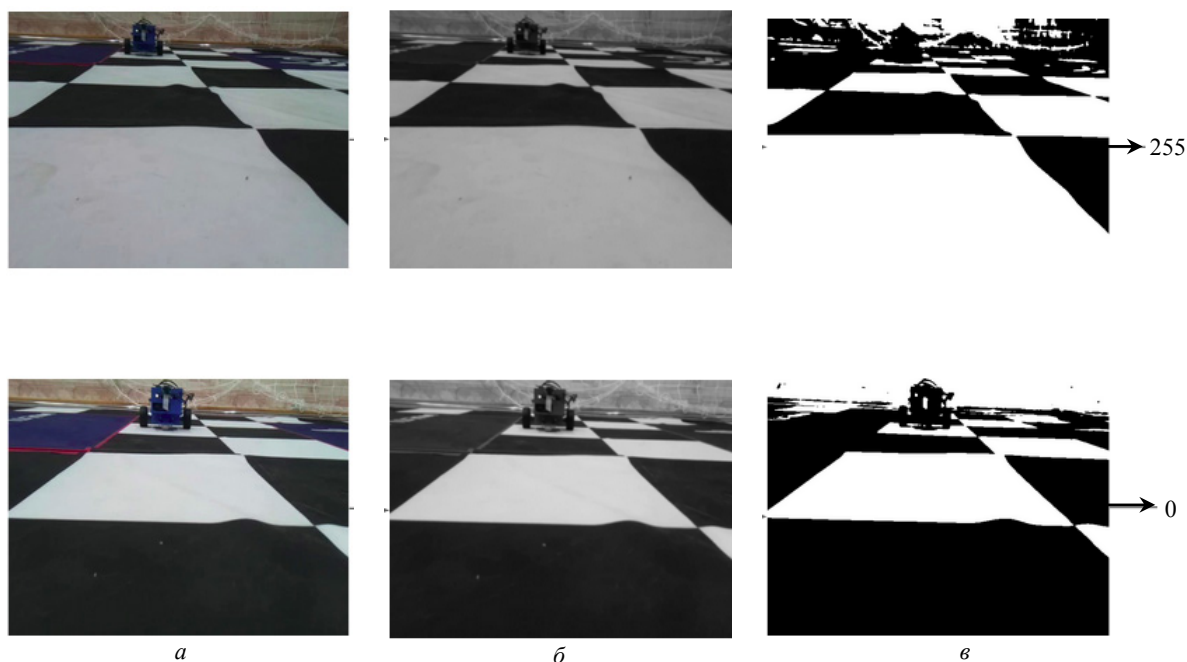


Рис. 16. Пример работы алгоритма определения цвета клетки: а – ЧБ + размытие по Гауссу; б – бинаризация Оцу; в – определение модального значения
 Fig. 16. Example of the algorithm for determining the color of a square: а – grayscale + Gaussian blur; б – Otsu's binarization; в – calculating the modal value

В результате применения предварительной обработки вероятность возникновения ошибки определения цвета клетки из-за наличия посторонних объектов на поле стенда сводится к минимуму. Так, на рис. 14 алгоритм без предобработки показывает неверный результат, указывая что цвет клетки черный. При включении в алгоритм предварительной обработки ошибка исправляется и модуль указывает на то, что цвет клетки – белый.

Детекция. Проведение экспериментов предполагает непрерывное движение всех БТС в различных направлениях. В таком случае возможны конфликтные ситуации в местах пересечения маршрутов. Неправильное принятие решения в такой ситуации может привести к столкновению между БТС. Для исключения такой возможности БТС должны уметь находить друг друга на изображении и использовать эту информацию при принятии решения.

Вследствие постоянно меняющейся окружающей среды, а также изменения угла обзора камеры, для реализации системы компьютерного зрения было решено использовать сверточные нейронные сети, которые позволят обобщить визуальные признаки БТС. Выбор архитектуры нейронной сети осложняется тем, что в качестве вычислителя в разрабатываемой системе используется маломощный одноплатный компьютер без графического процессора, а для удовлетворения предъявляемых к моделям БТС требований необходимо принятие решений в условиях реального времени.

После анализа существующих архитектур для поиска объектов, но наиболее подходящей оказалась архитектура EfficientDet-D0 [9]. Для реализации разрабатываемой системы компьютерного зрения, модель была обучена на 576 изображениях размером $640 \times 480 \times 3$. Тестовая выборка содержала 85 изображений. Параметры обучения модели приведены в табл. 2.

Табл. 2. Параметры обучения модели
Tab. 2. Training parameters of the model

Темп обучения	Оптимизатор	Размер батча	Число эпох
0.08	Adam [10]	4	40

Для повышения обобщающей способности модели обучающая выборка состояла из изображений БТС с различных ракурсов, в том числе с использованием методов аугментации

Эксперимент. Для проверки стабильности работы всей системы компьютерного зрения были проведены эксперименты при изменении различных внешних условий для выявления их влияния на работу каждого модуля в отдельности.

Для измерения влияния уровня освещенности на качество работы модуля коррекции было проведено несколько экспериментов при различных начальных условиях: при разном уровне освещенности (55, 325, 520 люкс), разных оттенках света (теплый и холодный) и при исключении предварительной обработки изображения.

В ходе эксперимента каждое транспортное средство проходило один и тот же маршрут от точки (9; 0) до точки (7; 4). Схема маршрута изображена на рис. 17.



Рис. 17. Схема с маршрутом
Fig. 17. Scheme of a planned route

На рис. 18 представлены графики угла отклонения направления движения БТС от прямолинейной траектории при различном освещении. Положительное направление вертикальной оси связано с коррекцией БТС в левую сторону, отрицательное – в правую. Идеальной считается траектория с нулевым углом отклонения, эта траектория совпадает с осью X . При правильной работе модуля коррекции отклонения должны компенсировать друг друга и таким образом выравнивать БТС. Периодическое изменение угла в левую и правую стороны заметно на всех графиках и все они колеблются около нуля. Также на всех графиках заметны резкие пики возрастания угла откло-

нения, они связаны с поворотом БТС и его последующей коррекцией. Таким образом, работа модуля коррекции не зависит от уровня освещения.

Наблюдение за частотой отклонений показывает, что БТС при любом уровне освещения чаще всего подворачивает в правую сторону, но при этом не выезжает за пределы намеченного маршрута. Гистограмма величины отклонения от прямолинейного движения изображена на рис. 19. Отклонение может быть связано с неравномерным распределением веса в конструкции БТС, а также невозможностью полной когерентной калибровки сервоприводов.

Независимость результатов работы модуля коррекции от уровня и оттенка освещения связана с наличием предварительной обработкой изобра-

жения. Этот факт подтверждается экспериментом, проведенным при аналогичных начальных условиях, но с отсутствием этапа предварительной обработки. Результаты эксперимента приведены на рис. 20. При проведении экспериментов во всех случаях БТС не достигло точки назначения, так как в процессе движения покинуло построенный маршрут из-за неправильной работы модуля коррекции. Погрешности работы модуля коррекции также видны на гистограмме рис. 21, из которой видно, что значения отклонений не сопоставляются ни с каким законом распределения и носят случайный характер.

Для проверки работы модуля позиционирования было проведено 5 экспериментов, в каждом из которых БТС двигался вперед с остановкой

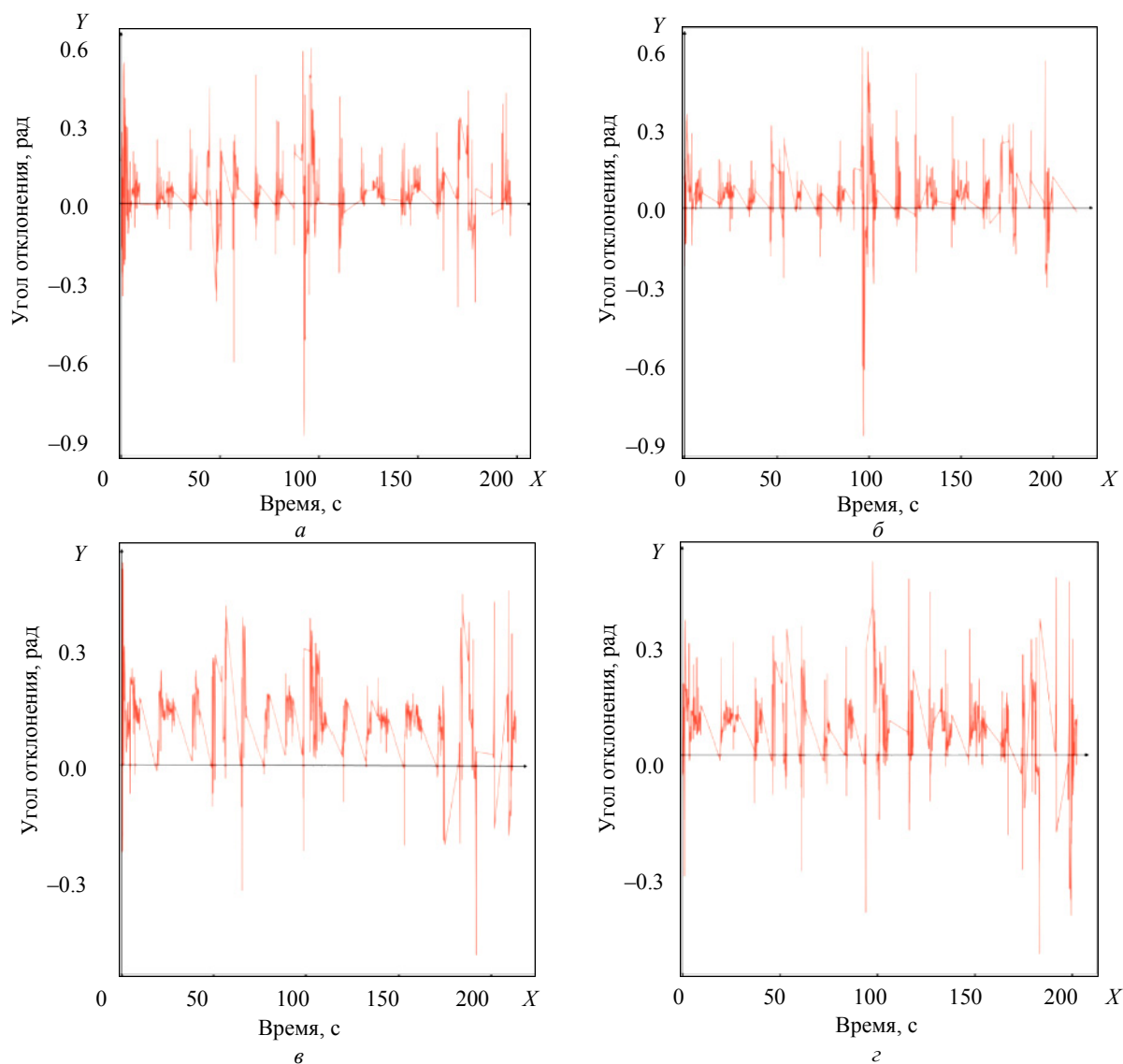


Рис. 18. Графики отклонения от прямолинейного движения: *a* – слабое освещение (55 люкс); *б* – холодный свет (325 люкс); *в* – теплый свет (325 люкс); *г* – полное освещение (520 люкс)

Fig. 18. Deviation from UV's rectilinear motion: *a* – low light (55 lx); *б* – cold light (325 lx); *в* – warm light (325 lx); *г* – full light (520 lx)

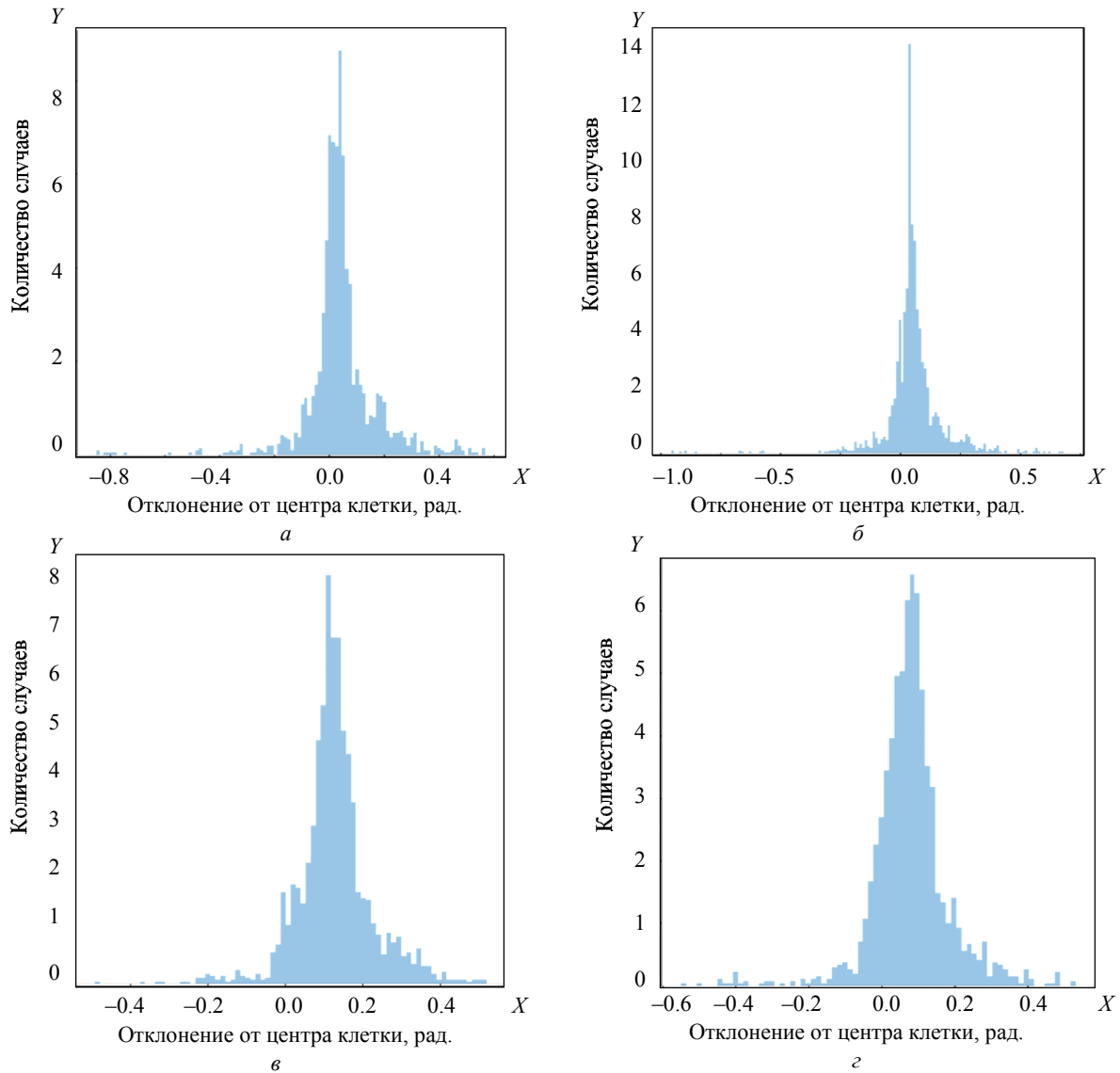


Рис. 19. Гистограмма отклонения от прямолинейного движения: а – слабое освещение (55 люкс); б – холодный свет (325 люкс); в – теплый свет (325 люкс); г – полное освещение (520 люкс)
 Fig. 19. Histogram of UV's deviation from a rectilinear motion: а – low light (55 lx); б – cold light (325 lx); в – warm light (325 lx); г – full light (520 lx)

при изменении координаты. Во время остановки измерялось отклонение положения БТС от края клетки. Результаты экспериментов представлены на рис. 22. На графике видно, что все отклонения находятся около нуля и много меньше размера клетки (30 см).

Проверка работоспособности модуля детекции связана с измерением его основных характеристик:

- точность работы: доля обнаруженных БТС, действительно являющаяся БТС;
- полнота предсказаний: доля БТС, идентифицированная системой компьютерного зрения как БТС;

– скорость работы: количество кадров в секунду, обработанное системой компьютерного зрения при работе в условиях реального времени.

Результаты работы модели на тестовой выборке приведены в табл. 3.

Анализ работы отдельных модулей показывает, что разработанная система компьютерного

Табл. 3. Результаты работы модели на тестовой выборке
 Tab. 3. Results demonstrated by the model on a test set

Метрика	Результат
Среднее значение точности	0.84
Среднее значение полноты	0.87
Скорость работы	3 кадра в секунду

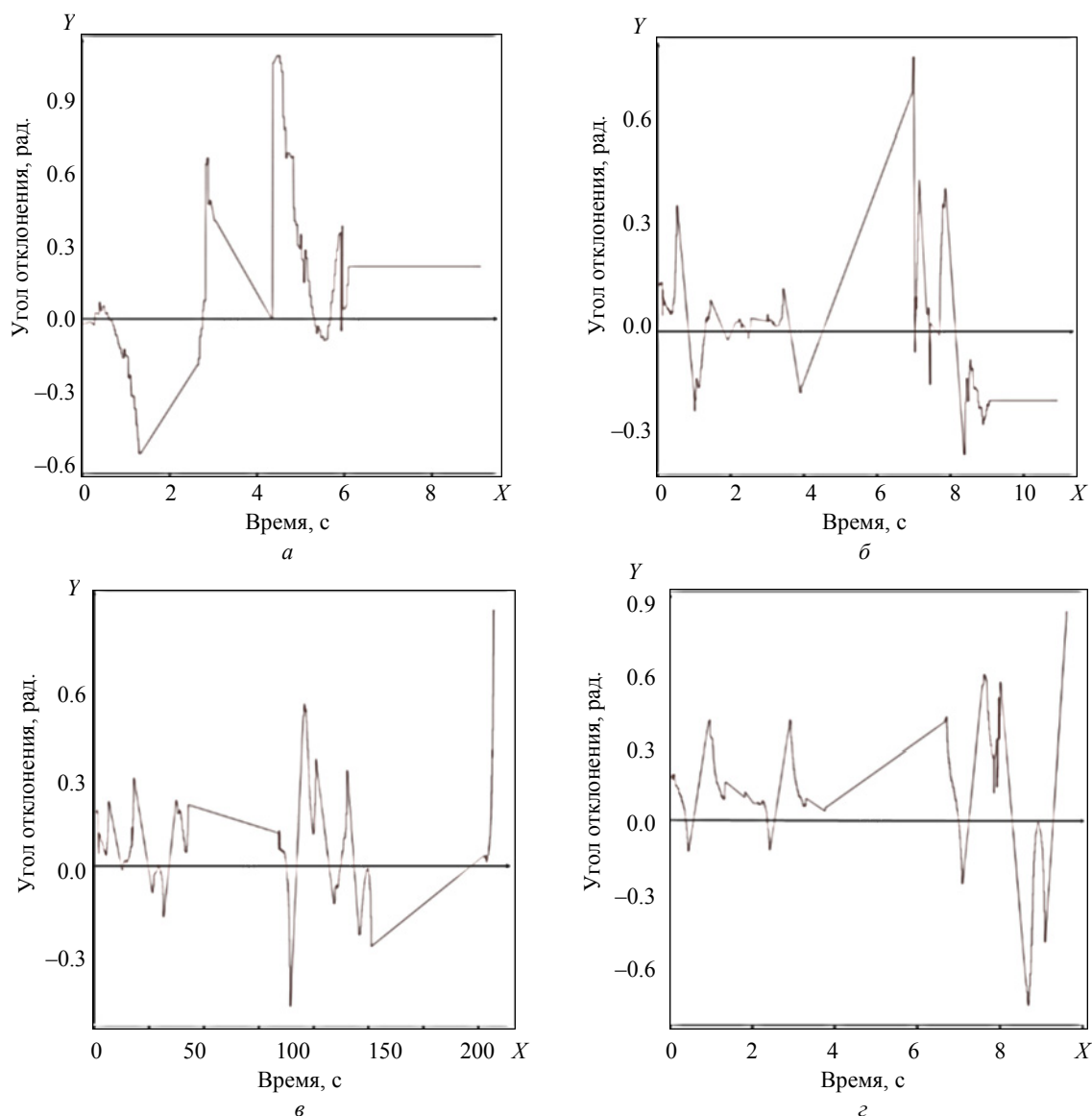


Рис. 20. Графики отклонения от прямолинейного движения (без этапа предварительной обработки):
а – слабое освещение (55 люкс); б – холодный свет (325 люкс); в – теплый свет (325 люкс);
г – полное освещение (520 люкс)
Fig. 20. Deviation from UV's rectilinear motion (without preprocessing): а – low light (55 lx); б – cold light (325 lx);
в – warm light (325 lx); г – full light (520 lx)

зрения остается устойчивой независимо от изменений внешней среды. При изменении уровня освещенности результат работы модулей не зависит от оттенка и яркости освещения.

Обсуждение результатов. Работа модулей системы удовлетворяет заявленным требованиям: модуль коррекции исключает выезд БТС за пределы маршрута; модуль локализации регистрирует изменение координат с допустимой погрешностью отклонения (10 %); модуль детекции работает с точностью 84 %; скорость работы модулей в совокупности, благодаря рациональному распределению вычислительных мощностей, удо-

влетворяет требованию обработки в реальном времени; кроссплатформенность позволяет интегрировать разработанное решение в большинство вычислительных систем.

Заключение. В данной статье рассмотрен процесс создания экспериментального стенда для исследования взаимодействия автономных беспилотных транспортных средств в структурной среде. Приведен аналитический обзор существующих решений в исследуемой области; для создания новой системы проведен анализ их преимуществ и недостатков, на котором основывалась разработка. Особенностью системы является

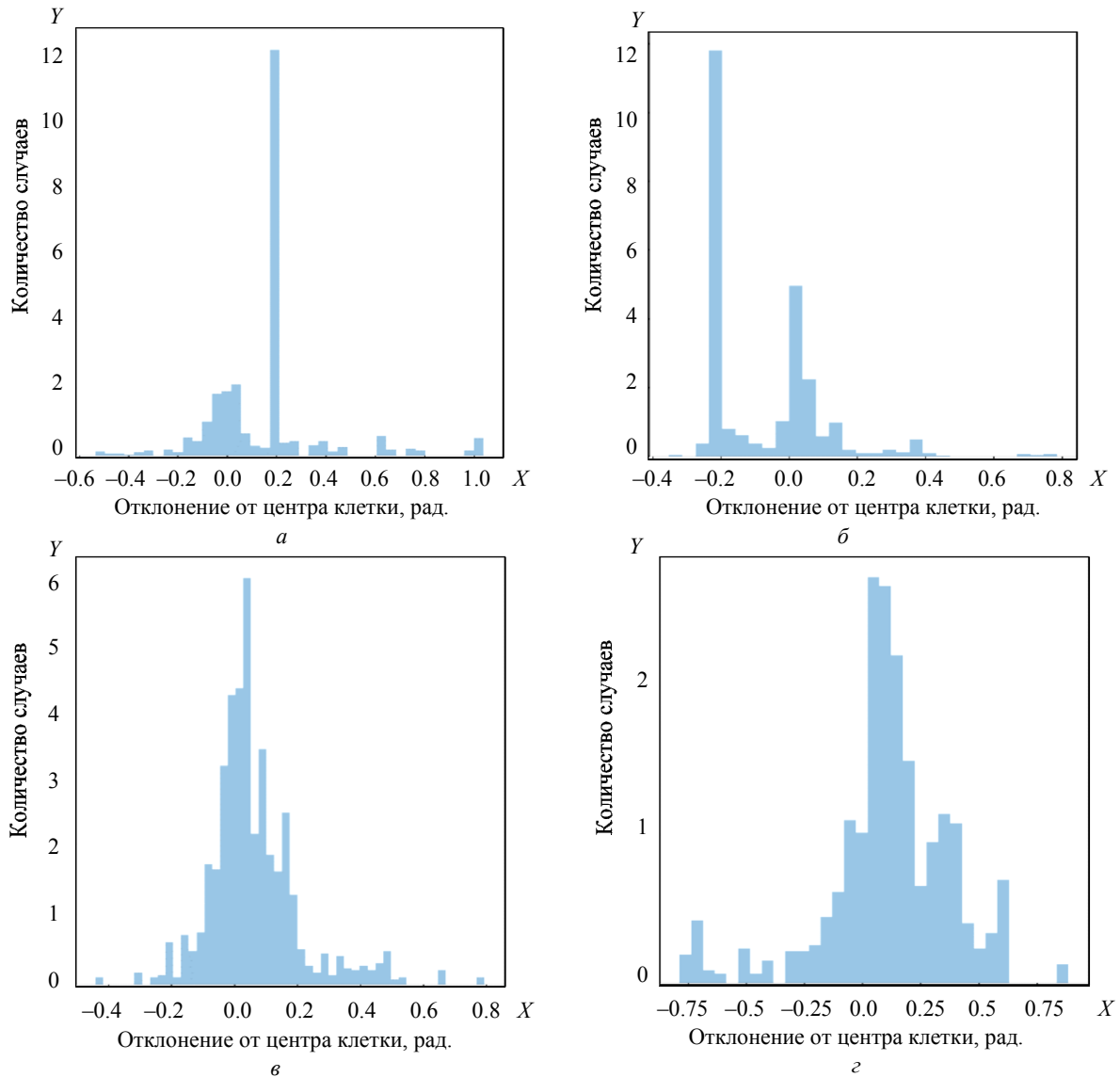


Рис. 21. Гистограмма отклонения от прямолинейного движения (при отсутствии предварительной обработки):
а – слабое освещение (55 люкс); б – холодный свет (325 люкс); в – теплый свет (325 люкс);
z – полное освещение (520 люкс)

Fig. 21. Histogram of UV's deviation from a rectilinear motion (without preprocessing): а – low light (55 lx);
б – cold light (325 lx); в – warm light (325 lx); z – full light (520 lx)

использование компьютерного зрения как основного источника информации о внешней среде без использования дополнительных датчиков. Согласно результатам проведенного эксперимента, разработанная система компьютерного зрения показала свою эффективность при изменении условий внешней среды: при наличии на макете посторонних объектов небольшого размера, его загрязнении, а также при изменении уровня и оттенка освещения. Используемая модульная архитектура, позволяет улучшать, добавлять и изменять функционал системы, расширяя или адаптируя ее под конкретную задачу. Разработанная система ком-

пьютерного зрения предназначена для интеграции в маломощные одноплатные компьютеры, что делает ее более дешевой по сравнению с аналогами, а также дает возможность ее адаптации на более мощных устройствах для увеличения эффективности.

Разработанный стенд позволяет проводить исследование и разработку алгоритмов коллективного управления мобильными робототехническими устройствами, изучать и тестировать методы управления умным городом. Таким образом, поставленная цель данного исследования достигнута.

В дальнейших исследованиях будут рассмотрены следующие задачи: масштабирование маке-

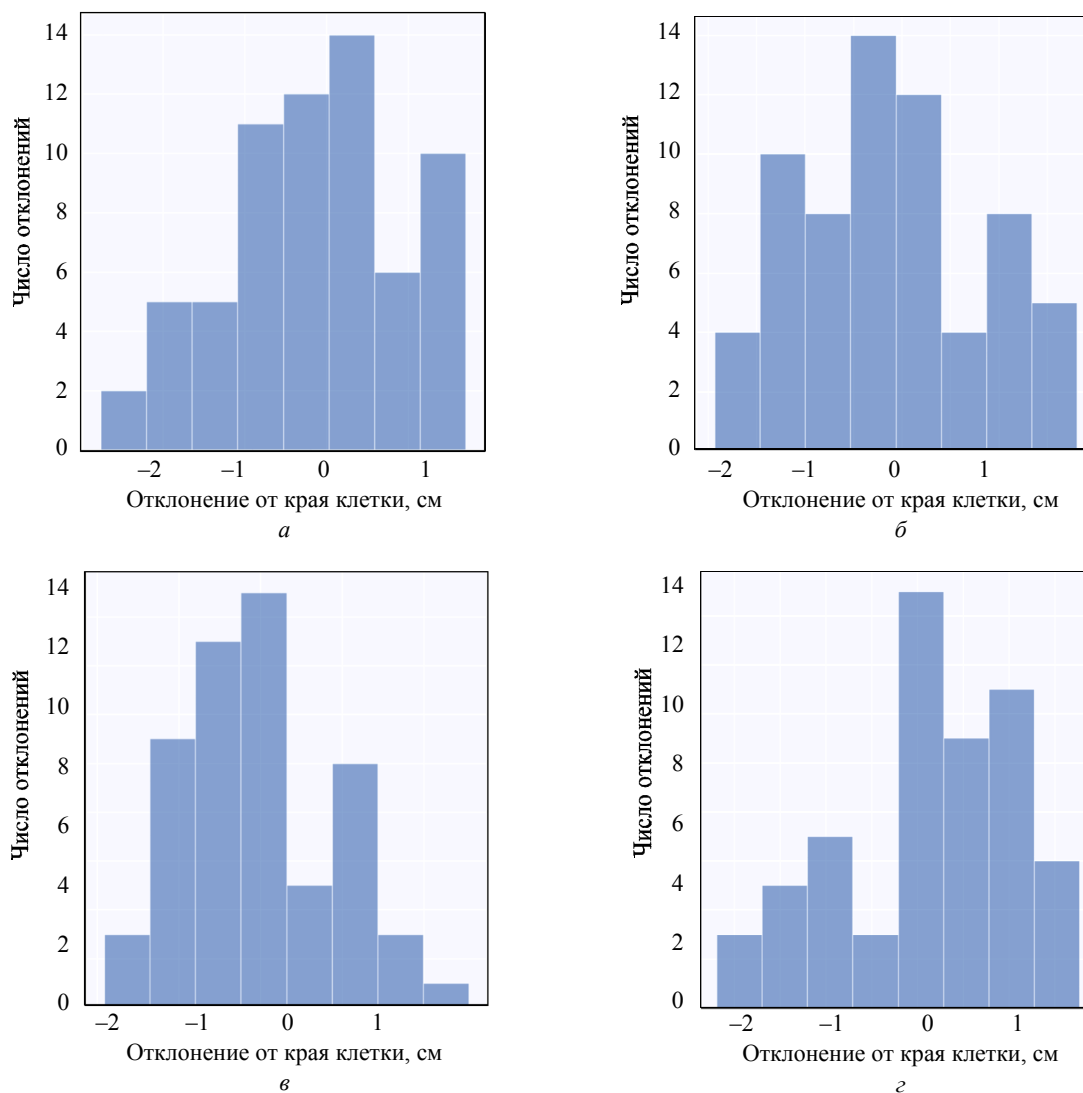


Рис. 22. Графики отклонений модуля позиционирования: а – слабое освещение; б – холодный свет; в – теплый свет; г – полное освещение

Fig. 22. UV's localization module error: а – low light; б – cold light; в – warm light; г – full light

та для исследования поведения системы в более сложной, приближенной к городской, внешней среде; создание комбинированного стенда, в котором

будет реализовано взаимодействие автономных робототехнических устройств различных модальностей: наземных и воздушных (летательных).

Список литературы

1. Безмен П. А., Бурцев А. П. Исследование сфер использования и классификация робототехнических систем // сб. тр. конф. «Поколение будущего: Взгляд молодых ученых-2018». Курск: ЗАО «Университетская книга», 2018. С. 94–97.
2. The evolution of robotic systems: Overview and its application in modern times / R. P. França, A. C. B. Monteiro, R. Arthu, Y. Iano // Safety, Security, and Reliability of Robotic Systems: Algorithms, Applications, and Technologies. Boca Raton: CRC Press, 2020. P. 1–20.
3. A survey of the recent architectures of deep convolutional neural networks / A. Khan, A. Sohail, U. Zahoor, A. S. Qureshi // Artificial intelligence rev. 2020. Vol. 53. P. 5455–5516. doi: 10.1007/s10462-020-09825-6.

4. Коробеев А. И., Чучаев А. И. Беспилотные транспортные средства: новые вызовы общественной безопасности // Lex russica. 2019. № 2 (147). С. 9–28.
5. Javaid A. Understanding Dijkstra's algorithm // SSRN Electronic J. 2013. P. 2340905. URL: https://www.academia.edu/5390703/Understanding_Dijkstra_Algorithm (дата обращения 05.04.2023). doi: 10.2139/ssrn.2340905.
6. Ding L., Goshtasby A. On the Canny edge detector // Pattern recognition. 2001. Vol. 34. № 3. P. 721–725.
7. Hu M. K. Visual Pattern Recognition by Moment Invariants // IEE Trans. Info. Theory. 1962. Vol. 8, № 2. P. 179–187.

8. Otsu N. A threshold selection method from gray-level histograms (англ.) // IEEE Trans. on Systems, Man, and Cybernetics. 1979. Vol. 9, № 1. P. 62–66. doi: 10.1109/TSMC.1979.4310076.

9. Tan M., Pang R., Le Q. V. Efficientdet: Scalable and efficient object detection // CVF Conf. on Computer Vision and Pattern Recognition. Seattle, WA, USA: IEEE,

2020. P. 10781–10790. URL: <https://ieeexplore.ieee.org/document/9156454> (дата обращения 05.04.2023). doi: 10.1109/CVPR42600.2020.01079.

10. Kingma D. P., Ba J. L. Adam: A method for stochastic optimization. arXiv: Learning, 2014. URL: <https://arxiv.org/pdf/1412.6980.pdf> (дата обращения 05.04.2023).

Информация об авторах

Белов Александр Михайлович – студент гр. 0306, кафедра вычислительной техники СПбГЭТУ «ЛЭТИ».

E-mail: Alexbelov.spb2@gmail.com

Беляев Павел Юрьевич – ассистент кафедры вычислительной техники СПбГЭТУ «ЛЭТИ».

E-mail: belyaev.edu@gmail.com

<https://orcid.org/0000-0001-5387-0622>

Виксин Илья Игоревич – канд. техн. наук, доцент кафедры вычислительной техники СПбГЭТУ «ЛЭТИ».

E-mail: wixnin@mail.ru

<https://orcid.org/0000-0002-3071-6937>

Ким Юлия Вячеславовна – младший научный сотрудник кафедры вычислительной техники СПбГЭТУ «ЛЭТИ».

E-mail: yulia1344@gmail.com

<https://orcid.org/0000-0002-6951-1875>

Радабольский Владислав Сергеевич – студент гр. 0308, кафедра вычислительной техники СПбГЭТУ «ЛЭТИ».

E-mail: radabolsky2017@gmail.com

Турушев Тимур Азаматович – студент гр. 0306, кафедра вычислительной техники СПбГЭТУ «ЛЭТИ».

E-mail: ta.turushev@gmail.com

Чупров Сергей Сергеевич – программист, кафедра вычислительной техники СПбГЭТУ «ЛЭТИ». Ассистент, Колледж вычислительной техники и информационных наук Голиcano, Рочестерский технологический институт. 20 Ломб Мемориал Драйв, Рочестер, штат Нью-Йорк, 14623, США.

E-mail: sc1723@rit.edu

<https://orcid.org/0000-0001-7081-8797>

References

1. Bezmen P. A., Burcev A. P. Issledovanie sfer ispol'zovaniya i klassifikacija robototekhnicheskikh sistem // sb. tr. konf. «Pokolenie budushhego: Vzglyad molodyh uchenyh-2018». Kursk: ZAO «Universitetskaja kniga», 2018. S. 94–97. (In Russ.).

2. The evolution of robotic systems: Overview and its application in modern times / R. P. França, A. C. B. Monteiro, R. Arthu, Y. Iano // Safety, Security, and Reliability of Robotic Systems: Algorithms, Applications, and Technologies. Boca Raton: CRC Press, 2020. P. 1–20.

3. A survey of the recent architectures of deep convolutional neural networks / A. Khan, A. Sohail, U. Zahoor, A. S. Qureshi // Artificial intelligence rev. 2020. Vol. 53. P. 5455–5516. doi: 10.1007/s10462-020-09825-6.

4. Korobeev A. I., Chuchaev A. I. Bepilotnye transportnye sredstva: novye vyzovy obshchestvennoj bezopasnosti // Lex russica. 2019. № 2 (147). S. 9–28. (In Russ.).

5. Javaid A. Understanding Dijkstra's algorithm // SSRN Electronic J. 2013. P. 2340905. URL: https://www.academia.edu/5390703/Understanding_Dijkstra_Algorithm (дата обращения 05.04.2023). doi: 10.2139/ssrn.2340905.

6. Ding L., Goshtasby A. On the Canny edge detector // Pattern recognition. 2001. Vol. 34. № 3. P. 721–725.

7. Hu M. K. Visual Pattern Recognition by Moment Invariants // IRE Trans. Info. Theory. 1962. Vol. 8, № 2. P. 179–187.

8. Otsu N. A threshold selection method from gray-level histograms (англ.) // IEEE Trans. on Systems, Man, and Cybernetics. 1979. Vol. 9, № 1. P. 62–66. doi: 10.1109/TSMC.1979.4310076.

9. Tan M., Pang R., Le Q. V. Efficientdet: Scalable and efficient object detection // CVF Conf. on Computer Vision and Pattern Recognition. Seattle, WA, USA: IEEE,

2020. P. 10781–10790. URL: <https://ieeexplore.ieee.org/document/9156454> (data obrashhenija 05.04.2023). doi: 10.1109/CVPR42600.2020.01079.

10. Kingma D. P., Ba J. L. Adam: A method for stochastic optimization. arXiv: Learning, 2014. URL: <https://arxiv.org/pdf/1412.6980.pdf> (data obrashhenija 05.04.2023).

Information about the authors

Alexander M. Belov – student gr. 0306, Department of Computer Engineering, Saint-Petersburg Electrotechnical University.

E-mail: Alexbelov.spb2@gmail.com

Pavel Yu. Belyaev – assistant of the Department of Computer Engineering, Saint Petersburg State Electrotechnical University.

E-mail: belyaev.edu@gmail.com

<https://orcid.org/0000-0001-5387-0622>

Ilya I. Viksnin – Cand. Sci. (Eng.), Associate Professor of the Department of Computer Engineering, Saint Petersburg State Electrotechnical University.

E-mail: wixnin@mail.ru

<https://orcid.org/0000-0002-3071-6937>

Yulia V. Kim – junior research assistant of the Department of Computer Engineering, Saint Petersburg State Electrotechnical University.

E-mail: yulia1344@gmail.com

<https://orcid.org/0000-0002-6951-1875>

Vladislav S. Radabolsky – student gr. 0308, Department of Computer Engineering, Saint Petersburg State Electrotechnical University.

E-mail: radabolsky2017@gmail.com

Timur A. Turushev – student gr. 0306, Department of Computer Engineering, Saint Petersburg State Electrotechnical University.

E-mail: ta.turushev@gmail.com

Sergey S. Chuprov – software developer, Department of Computer Engineering, Saint Petersburg Electrotechnical University; Assistant, Golisano College of Computing and Information Sciences, Rochester Institute of Technology, Rochester, NY, 14623, USA.

E-mail: sc1723@rit.edu

<https://orcid.org/0000-0001-7081-8797>

Статья поступила в редакцию 17.04.2023; принята к публикации после рецензирования 03.05.2023; опубликована онлайн 23.06.2023.

Submitted 17.04.2023; accepted 03.05.2023; published online 23.06.2023.
