

**РОБОТЫ, МЕХАТРОНИКА И РОБОТОТЕХНИЧЕСКИЕ СИСТЕМЫ / ROBOTS, MECHATRONICS AND
ROBOTIC SYSTEMS**

DOI: <https://doi.org/10.60797/IRJ.2024.143.156>

**ОПРЕДЕЛЕНИЕ ПОЛОЖЕНИЯ РАБОЧЕГО ИНСТРУМЕНТА РОБОТА-МАНИПУЛЯТОРА С ПОМОЩЬЮ
КАМЕРЫ И МАШИННОГО ЗРЕНИЯ**

Научная статья

Рублева Е.А.^{1,*}, Мудрич А.Б.²

¹ORCID : 0000-0003-3978-1484;

¹Уральский федеральный университет имени первого Президента России Б.Н. Ельцина, Екатеринбург, Российская Федерация

²Национальный исследовательский университет ИТМО, Санкт-Петербург, Российская Федерация

* Корреспондирующий автор (elena[at]erubleva.ru)

Аннотация

В статье рассматривается проблематика управления роботом-манипулятором и определения координат объектов в пространстве с использованием технологий машинного зрения. Главная задача исследования – разработка метода позиционирования рабочего инструмента манипулятора через анализ изображения камеры. Для описания модели робота и расчета движений применяется кинематический подход с параметрами Денавита-Хартенберга. В качестве практической реализации для определения отклонений, корректировки положения рабочего инструмента и написания программного кода используется OpenCV и Robot Operating System (ROS). Результаты обработки изображений с камеры выводятся в пользовательский интерфейс. Эксперименты проводились на небольшом настольном роботоманипуляторе с параллельным механизмом. В конечном итоге получается рабочее приложение, которое способно улучшить точность позиционирования роботизированных систем. Тем не менее для практического использования системы необходимо сначала провести калибровку оборудования и адаптировать камеры к условиям освещения.

Ключевые слова: обработка изображений, машинное зрение, управление роботом-манипулятором, OpenCV.

**DETERMINATION OF THE POSITION OF THE ROBOT MANIPULATOR TOOL USING A CAMERA AND
MACHINE VISION**

Research article

Rubleva Y.A.^{1,*}, Mudrich A.B.²

¹ORCID : 0000-0003-3978-1484;

¹Ural Federal University named after the first President of Russia B.N. Yeltsin, Ekaterinburg, Russian Federation

²ITMO University, Saint-Petersburg, Russian Federation

* Corresponding author (elena[at]erubleva.ru)

Abstract

The article examines the problem of controlling a robot manipulator and determining the coordinates of objects in space using machine vision technologies. The main task of the research is to develop a method of positioning the manipulator working tool through analysing the camera image. A kinematic approach with Denavit-Hartenberg parameters is used to describe the robot model and calculate the motions. As a practical implementation, OpenCV and Robot Operating System (ROS) are used for defining deviations, correcting the position of the working tool and writing programme code. The results of the camera image processing are displayed in the user interface. Experiments were conducted on a small desktop robot manipulator with parallel mechanism. The final result is a working application that can improve the positioning accuracy of robotic systems. However, for practical use of the system, the equipment must first be calibrated, and the cameras adapted to the lighting conditions.

Keywords: image processing, machine vision, robot manipulator control, OpenCV.

Введение

Для решения различных задач движения робототехнических комплексов и определения координат предметов в пространстве в современных информационных науках наиболее эффективно применяются алгоритмы машинного зрения и алгоритмы глубокого машинного обучения. Они дают возможность распознать текущее положение предмета в пространстве, предсказать его дальнейшее движение и оптимальную траекторию перемещения. В исследовании алгоритмы машинного зрения применяются для решения практической задачи определения текущего положения рабочего инструмента робота-манипулятора после завершения роботом движения и возврата в исходную позицию. В случае наличия несоответствия рассчитывается отклонение по осям x , y , z для перемещения рабочего органа в правильное положение. Проблема возможного расхождения между фактическим положением рабочего органа и планируемым актуальна для проверки точности работы любого робота-манипулятора, не имеющего альтернативных механизмов, реализующих обратную связь (например, энкодеров на двигателях).

Методы исследования

2.1. Цели и задачи исследования

Целью исследования является разработка системы корректировки положения рабочего инструмента робота-манипулятора с использованием камеры и алгоритмов машинного зрения. Конструкция робота представляет собой разомкнутую кинематическую цепь с параллельным механизмом. Для перемещения звеньев в пространстве используются шаговые двигатели стандарта NEMA17 с планетарным редуктором.

Для определения фактического положения рабочего инструмента робота-манипулятора и вычисления отклонения текущего положения от исходного на основе полученных координат необходимо:

Установить и подключить камеру, с которой будет считываться изображение;

Обработать изображение с камеры;

Исследовать библиотеку машинного зрения OpenCV [4];

Разработать методику определения положения рабочего инструмента робота-манипулятора с помощью OpenCV;

Выполнить расчеты и вывести результаты в пользовательский интерфейс;

Протестировать написанное программное обеспечение.

Исследование проводилось на модели робота-манипулятора (рисунок 1). Для написания программного обеспечения использовался фреймворк Robot Operating System (ROS) [3], [6].

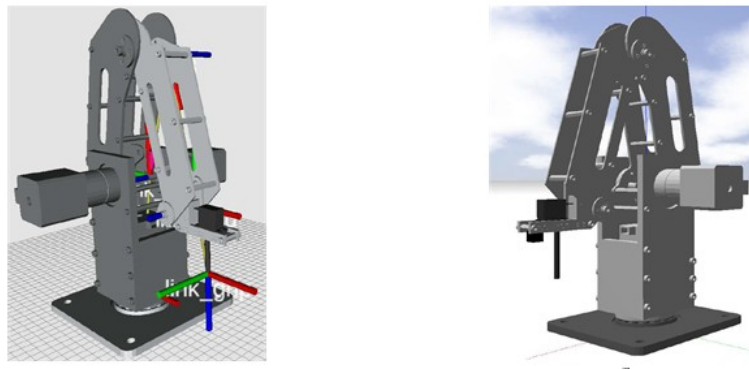


Рисунок 1 - Симуляция модели
DOI: <https://doi.org/10.60797/IRJ.2024.143.156.1>

Примечание: а – в визуализаторе rViz, б – в симуляторе Gazebo [1]

2.2. Кинематическая схема модели робота-манипулятора

Кинематическая схема модели робота-манипулятора представлена на рисунке 2 [7]. Все звенья модели являются вращательными. За счет привязки координат к звеньям манипулятора используются параметры Денавита-Хартенберга: то есть вместо 6 координат (3 по положению и 3 по ориентации) нужны только 4 (таблица 1).

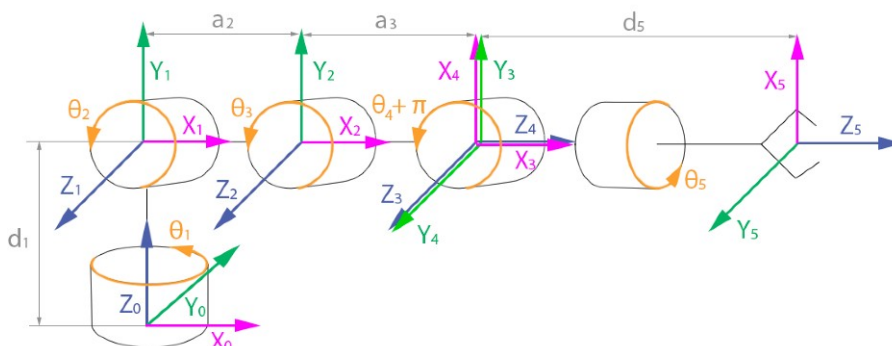


Рисунок 2 - Кинематическая схема модели робота-манипулятора
DOI: <https://doi.org/10.60797/IRJ.2024.143.156.2>

Таблица 1 - Параметры Денавита-Хартенберга

DOI: <https://doi.org/10.60797/IRJ.2024.143.156.3>

Link, i	a_i	α_i	d_i	θ_i
1	0	$\pi/2$	d_1	θ_{i1}

2	a2	0	0	theta ₂
3	a3	0	0	theta ₃
4	0	$\pi/2$	0	theta ₄ + $\pi/2$
5	0	0	d ₅	theta ₅

Глобальная система координат модели изображена на рисунке 3.

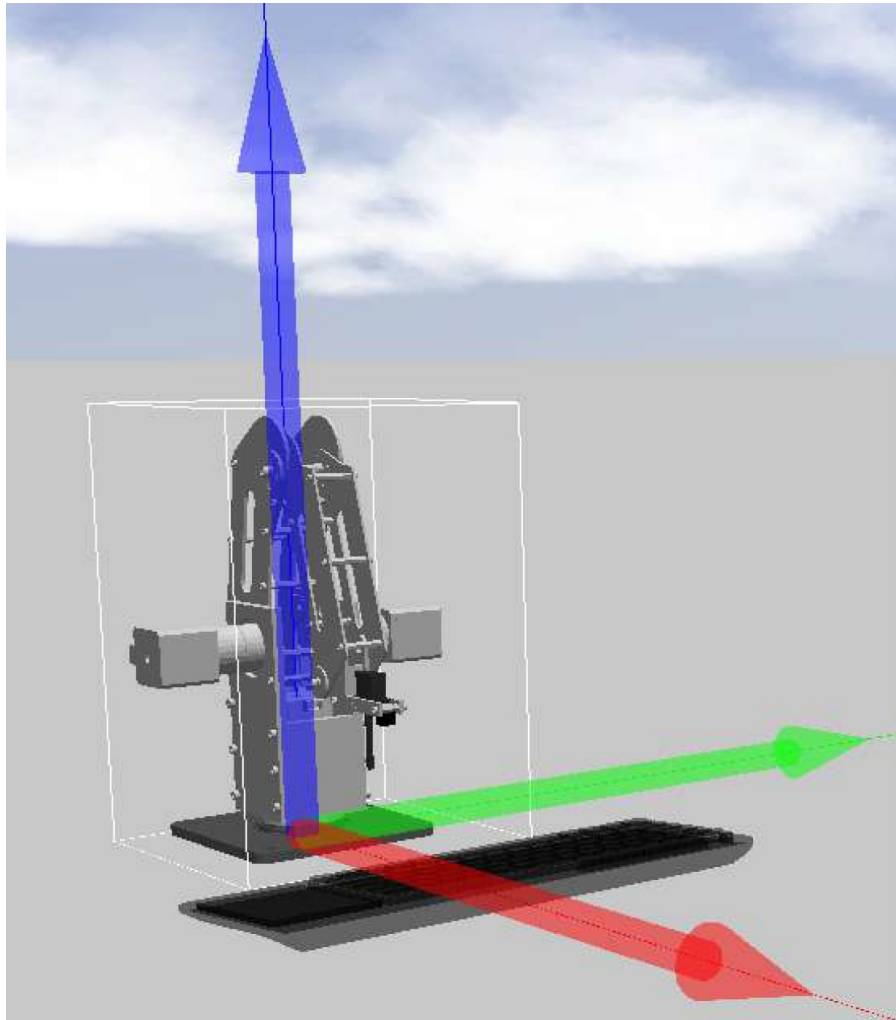


Рисунок 3 - Кинематическая схема модели робота-манипулятора
 DOI: <https://doi.org/10.60797/IRJ.2024.143.156.4>

Смещение конечной системы координат относительно базовой произведено с помощью матричных однородных преобразований.

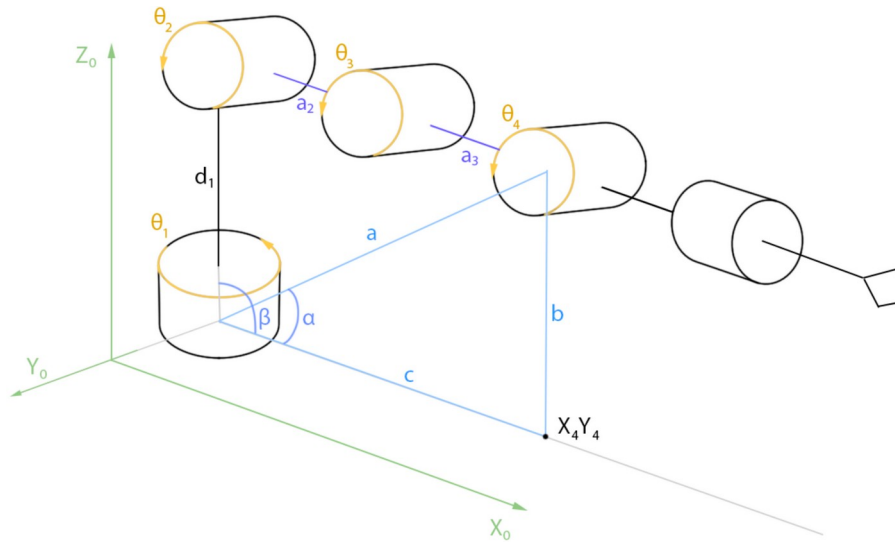


Рисунок 4 - Геометрическая схема робота
DOI: <https://doi.org/10.60797/IRJ.2024.143.156.5>

Решение задачи обратной кинематики можно разделить на 2 части:

решение задачи обратной кинематики по положению;

решение задачи обратной кинематики по ориентации [10].

В рамках исследования была решена только одна задача обратной кинематики по положению – задача поиска обобщенных координат при известных координатах положения рабочего инструмента с помощью геометрической схемы модели робота (рисунок 4). Это позволило вычислить углы между звеньями каждой кинематической пары [5], [8].

2.3. Техническое зрение

В проекте используется функциональная схема аппаратной части робототехнического комплекса с сервером данных и системой технического зрения [11].

Для работы с изображением была произведена бинаризация исходного изображения: уменьшено количество информации, полученной с камеры. Исходное изображение с камеры до преобразований по умолчанию сохраняется в формате RGB, неудобном для последующего использования и обработки. Для очищения изображения от фона производится его преобразование в формат HSV (Hue, Saturation, Value – тон, насыщенность, значение) и определяются верхний и нижний пределы заданного цвета. На основе полученных значений проводится последующая обработка изображения.

Основные результаты

В качестве идентификатора положения рабочего инструмента робота-манипулятора служит специальный маркер, установленный на верхний сервопривод. Он позволяет после обработки изображения по цвету определять местоположение отслеживаемого объекта.

Для получения информации о текущем положении фиксируемого объекта в модель были добавлены датчики камеры. Пример получаемых с камер изображений представлен на рисунке 5.

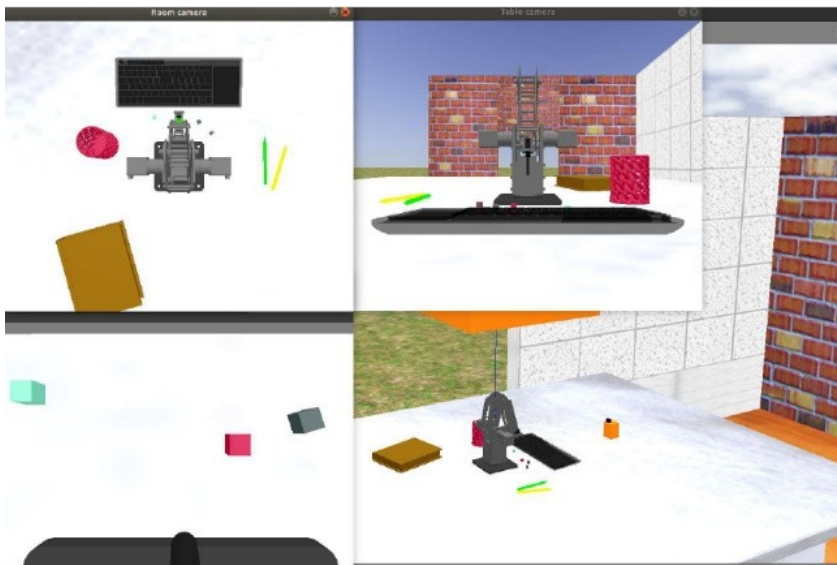


Рисунок 5 - Изображения с камер в симуляторе
DOI: <https://doi.org/10.60797/IRJ.2024.143.156.6>

Обработка исходных изображений с датчиков камер происходит поэтапно и включает в себя следующие фазы (рисунок 6):

1. Получение оригинального изображения;
2. Конвертация полученного изображения в формат в HSV;
3. Наложение маски на маркер, установленный на месте крепления рабочего инструмента;
4. Наложение линии, соединяющей начальное и конечное положение рабочего инструмента.

Основной датчик камеры, с помощью которого происходит фиксация, расположен сверху над центром основания модели робота. Данные о размерах основных предметов внешней среды и самого робота-манипулятора (размеры робота, стола, высоту крепления камеры и т.д.) позволяют найти отклонение точки по оси x , y .

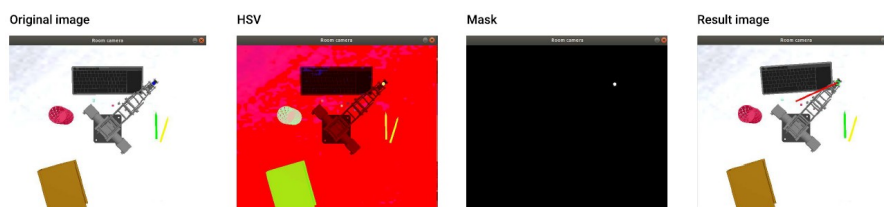


Рисунок 6 - Поэтапная обработка изображения
DOI: <https://doi.org/10.60797/IRJ.2024.143.156.7>

При расчете отклонения по оси Z используется изображение с камеры, которая закреплена фронтально перед роботом. Для этого используется алгоритм детектирования объекта по цвету, описанный в предыдущем разделе (рисунок 6). Неотъемлемой частью решения задачи определения координат рабочего инструмента являются правила формирования изображения с камеры с геометрической точки зрения.

На рисунке 7 на схеме справа изображена мировая система координат комнаты (показана оранжевым цветом) и система координат камеры (показана красным цветом). Системы координат связаны 6 параметрами: 3-мя для перемещения и 3-мя для вращения. На снимке камеры необходимо найти координаты точки P (нас интересует только Z – высота). Зная местоположение камеры и местоположение рабочего инструмента, которое найдено по камере сверху, рассчитывается высота точки.

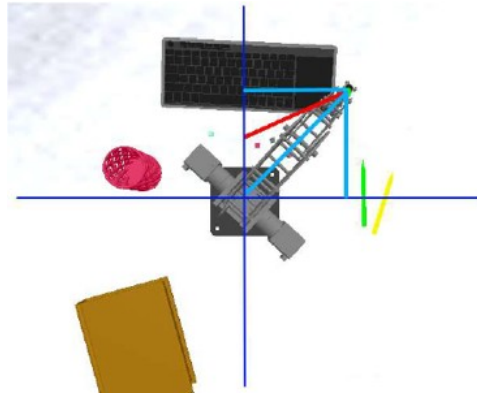


Рисунок 7 - Геометрическое решение задачи нахождения местоположения рабочего инструмента
DOI: <https://doi.org/10.60797/IRJ.2024.143.156.8>

Изображения с камеры используются для построения карты глубины. Эта информация необходима для определения расстояния до объекта.

Чтобы построить карту глубины, в модели необходимо разместить еще одну камеру, т.к. нужно получать изображение с 2-х ракурсов. При трансляции картинки с камеры трехмерный объект на изображении проецируется из трехмерного пространства в двумерное пространство. Это называется планарной проекцией. Проблема в том, что при этой операции теряется информация о глубине. И для восстановления этих данных недостаточно использовать одно изображение. Двух снимков вполне достаточно, потому что лучи, исходящие из точки C1 и C2 пересекаются в одной единственной точке X (см. рисунок 9b) [9]. Этот процесс называется триангуляцией точки X. Для восстановления глубины необходимо знать положение камер и соответствие точек на изображениях.

На рисунке 9а показана связь между глубиной Z и диспаратностью, то есть разностью точки на изображениях, представленных с разных ракурсов [2].

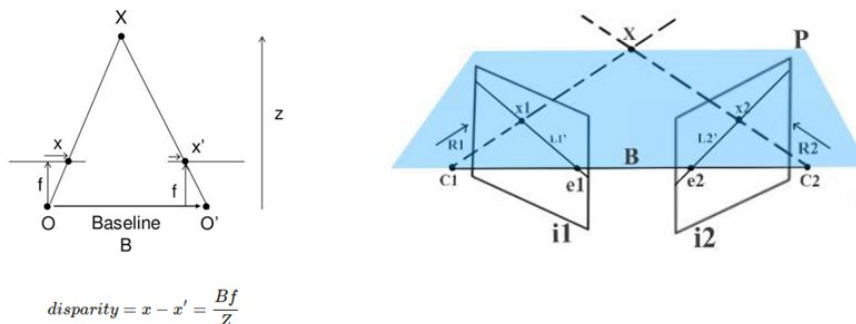


Рисунок 8 - Симуляция модели
DOI: <https://doi.org/10.60797/IRJ.2024.143.156.9>

Примечание: слева – диспаратность изображения; справа – геометрия диспаратности

Чтобы получить изображение, с которым можно было бы работать, необходимо найти взаимосвязь между значениями глубины и диспаратности для настройки стереокамеры.

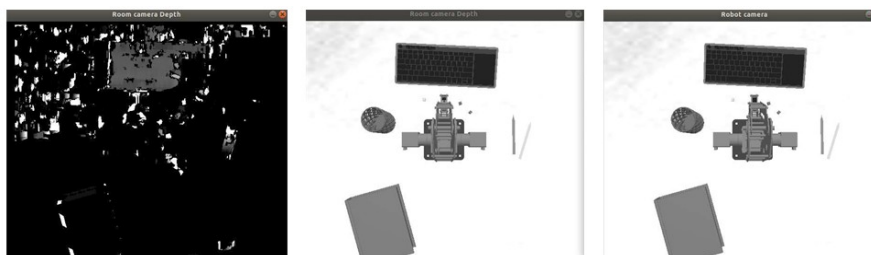


Рисунок 9 - Построение карты глубины
DOI: <https://doi.org/10.60797/IRJ.2024.143.156.10>

Примечание: слева – карта глубины; по центру – изображение с первой камеры; справа – изображение со второй камеры

Пример полученной карты глубины представлен на рисунке 10а. Изображения, полученные с разных ракурсов, и на основании которых построена карта глубины, представлены на рисунках 10б и 10в.

В качестве альтернативного решения задачи рассматривается разработка сверхточной нейронной сети. Первая задача, которая ставится для нейросети – это задача классификации. Необходимо определить по снимку с камеры – находится ли инструмент робота-манипулятора в исходном положении или нет. Это – задача будущих исследований.

Заключение

Подводя итоги проведенного исследования, можно кратко описать промежуточные этапы и достигнутые результаты:

1. Подготовлена модель для симуляции;
2. Найдено оптимальное положение камер для получения информации о положении рабочего инструмента робота-манипулятора;
3. Подготовлен алгоритм для идентификации текущего положения рабочего инструмента робота манипулятора;
4. Разработан алгоритм возвращения модели в исходную позицию;
5. Построена карта глубины;
6. Информация выведена в пользовательский интерфейс.

Полученные результаты можно использовать на практике. Для этого необходимо откалибровать камеру, т.к. картинка с камеры может быть с искажениями, а также нужно учитывать уровень освещения в помещении.

Конфликт интересов

Не указан.

Conflict of Interest

None declared.

Рецензия

Все статьи проходят рецензирование. Но рецензент или автор статьи предпочли не публиковать рецензию к этой статье в открытом доступе. Рецензия может быть предоставлена компетентным органам по запросу.

Review

All articles are peer-reviewed. But the reviewer or the author of the article chose not to publish a review of this article in the public domain. The review can be provided to the competent authorities upon request.

Список литературы / References

1. Gazebo (Robot simulation): documentation for Gazebo. — USA. — URL: <https://gazebosim.org/> (accessed: 15.07.2023)
2. Depth Map from Stereo Images: documentation for OpenCV. — USA. — URL: https://docs.opencv.org/4.x/dd/d53/tutorial_py_depthmap.html (accessed: 24.08.2023)
3. O’Kane J.M. A Gentle Introduction to ROS / J. M. O’Kane. — Columbia: University of South Carolina, 2014. — 156 p.
4. OpenCV: documentation for OpenCV. — USA. — URL: <https://opencv.org/> (accessed: 12.09.2023)
5. Park F.C. Introduction to robotics. Mechanics, planning, and control / F.C. Park, K. M. Lynch. — USA: Pearson Education International, 2016. — 400 p.
6. ROS — Robot Operating System: documentation for ROS. — USA. — URL: <https://www.ros.org/> (accessed: 15.07.2023)
7. Rubleva E.A. Kinematic Model of a Desktop Robot Manipulator with 5 Degrees of Freedom / E.A. Rubleva, A.B. Mudrich // Journal of Physics: Conference Series. — vol. 2096. — № 1. — 012178
8. Spong W.M. Robot Modelling and Control / W.M. Spong, S. Hutchinson, M. Vidyasagar. — New York: Wiley. — 2 ed. — 2020. — 407 p.
9. Stereo Camera Depth Estimation With OpenCV (Python/C++) // LearnOpenCV blog. — USA. — URL: <https://learnopencv.com/depth-perception-using-stereo-camera-python-c/> (accessed: 24.08.2023)
10. Zhang J. Performance optimization and implementation of evolutionary inverse kinematics in ROS / J. Zhang. — Hamburg: Hamburg University, 2017. — 69 p.
11. Борисов О.И. Методы управления робототехническими приложениями: учебное пособие / О.И. Борисов, В.С. Громов, А.А. Пыркин. — Санкт-Петербург: Университет ИТМО, 2016. — 110 с.
12. Жавнер В.Л. Мехатронные системы: учебное пособие / В.Л. Жавнер, А.Б. Смирнов. — Санкт-Петербург: Издательство Политехнического университета, 2011. — 131 с.
13. Разработка управляющих программ промышленных роботов / А.С. Климчик, Р.И. Гомолицкий, Ф.В. Фурман [и др.] — Минск: Белорусский государственный университет информатики и радиоэлектроники, 2008. — 131 с.
14. Колюбин С.А. Динамика робототехнических систем: учебное пособие / С.А. Колюбин. — Санкт-Петербург: Университет ИТМО, 2017. — 119 с.

Список литературы на английском языке / References in English

1. Gazebo (Robot simulation): documentation for Gazebo. — USA. — URL: <https://gazebosim.org/> (accessed: 15.07.2023)

2. Depth Map from Stereo Images: documentation for OpenCV. — USA. — URL: https://docs.opencv.org/4.x/dd/d53/tutorial_py_depthmap.html (accessed: 24.08.2023)
3. O’Kane J.M. A Gentle Introduction to ROS / J. M. O’Kane. — Columbia: University of South Carolina, 2014. — 156 p.
4. OpenCV: documentation for OpenCV. — USA. — URL: <https://opencv.org/> (accessed: 12.09.2023)
5. Park F.C. Introduction to robotics. Mechanics, planning, and control / F.C. Park, K. M. Lynch. — USA: Pearson Education International, 2016. — 400 p.
6. ROS — Robot Operating System: documentation for ROS. — USA. — URL: <https://www.ros.org/> (accessed: 15.07.2023)
7. Rubleva E.A. Kinematic Model of a Desktop Robot Manipulator with 5 Degrees of Freedom / E.A. Rubleva, A.B. Mudrich // Journal of Physics: Conference Series. — vol. 2096. — № 1. — 012178
8. Spong W.M. Robot Modelling and Control / W.M. Spong, S. Hutchinson, M. Vidyasagar. — New York: Wiley. — 2 ed. — 2020. — 407 p.
9. Stereo Camera Depth Estimation With OpenCV (Python/C++) // LearnOpenCV blog. — USA. — URL: <https://learnopencv.com/depth-perception-using-stereo-camera-python-c/> (accessed: 24.08.2023)
10. Zhang J. Performance optimization and implementation of evolutionary inverse kinematics in ROS / J. Zhang. — Hamburg: Hamburg University, 2017. — 69 p.
11. Borisov O.I. Metody upravleniya robototekhnicheskimi prilozheniyami: uchebnoe posobie [Methods of controlling robotic applications: study guide] / O.I. Borisov, V.S. Gromov, A.A. Pyrkin. — St. Petersburg: University ITMO, 2016. — 110 p. [in Russian]
12. ZHavner V.L. Mekhatronnye sistemy: uchebnoe posobie [Mechatronic systems: study guide] / V.L. ZHavner, A.B. Smirnov. — St. Petersburg: Polytechnic University Press, 2011. — 131 p. [in Russian]
13. Razrabotka upravlyayushchih programm promyshlennyh robotov [Development of control programs for industrial robots] / A.S. Klimchik, R.I. Gomolickij, F.V. Furman [et al.] — Minsk: Belarusian State University of Informatics and Radioelectronics, 2008. — 131 p. [in Russian]
14. Kolyubin S.A. Dinamika robototekhnicheskikh sistem: uchebnoe posobie [Dynamics of robotic systems: study guide] / S.A. Kolyubin. — St. Petersburg: University ITMO, 2017. — 119 p. [in Russian]