

МОБИЛЬНЫЙ МАНИПУЛЯЦИОННЫЙ РОБОТ ДЛЯ МЕЖДУНАРОДНЫХ СОРЕВНОВАНИЙ ПО РОБОТОТЕХНИКЕ**Р.О. Анисимов**

rodion_anisimov@mail.ru

SPIN-код: 2973-6260

В.С. Бакаев

bakaev.98@bk.ru

SPIN-код: 4166-4544

Т.Б. Бахов

tamelan.baxov@mail.ru

SPIN-код: 7933-2300

Н.В. Голобурдин

nickolaygoloburdin@mail.ru

SPIN-код: 4454-8951

А.М. Марчук

Marchuk.sasha2010@yandex.ru

SPIN-код: 9253-5492

Н.А. Мостаков

nikrus333@gmail.com

SPIN-код: 5393-7773

МГТУ им. Н.Э. Баумана, Москва, Российская Федерация**Аннотация**

Обобщен опыт участия команды студентов МГТУ им. Н.Э. Баумана в чемпионате по робототехнике RoboCup в номинации RoboCup@Work. Этот чемпионат является одним из самых престижных студенческих мероприятий, в которых принимают участие университеты со всего мира. Данная номинация нацелена на имитацию действий робота в условиях складского помещения. Основные задачи лежат в области навигации и манипулирования объектами. Приведено описание основных подсистем мобильного манипуляционного робота: навигационной системы, системы технического зрения, основанной на комбинации алгоритмов глубокого обучения и классических методов обработки изображений, манипуляционной системы и конечного автомата.

Ключевые слова

Мобильный манипуляционный робот, RoboCup, RoboCup@Work, Bauman Robotics Club, конечный автомат, навигация, техническое зрение, RealSense, манипулятор

Поступила в редакцию 09.11.2020

© МГТУ им. Н.Э. Баумана, 2020

Введение. RoboCup представляет собой некоммерческий, научный и культурный проект по продвижению искусственного интеллекта, робототехники и других связанных областей науки и техники посредством организации и проведения робототехнических соревнований. История международных соревнований начинается с 1997 г. В этом году финал чемпионата RoboCup Азиатско-Тихоокеанского региона проводился 6–11 ноября 2019 г. в Москве на территории МГТУ им. Н.Э. Баумана. Россию в номинации RoboCup@Work представляла команда Bauman Robotics Club (BRC), состоящая из студентов 4-го курса кафедры СМ7 и представившая для участия в RoboCup@Work мобильный манипуляционный робот (ММР) собственной конструкции.

Целью соревнований является использование роботов в сценариях, реально существующих в промышленности. Примерами таких сценариев являются загрузка или разгрузка контейнеров с различными объектами, погрузка и доставка деталей, а также гибкое и динамическое планирование процессов сборки и транспортировки с участием нескольких агентов (людей, роботов и машин). Сценарии RoboCup@Work нацелены на сложные, в основном нерешенные проблемы в области робототехники, искусственного интеллекта и передовых компьютерных наук, в частности в области восприятия, планирования пути и движений, манипулирования объектами, планирования, обучения и адаптации, а также вероятностного моделирования [1].

В рамках данных соревнований необходимо было разработать ММР для решения следующих задач:

- навигации (достижения в статической среде указанных в задании позиций с определенной ориентацией без столкновений);
- манипуляции (захвата и переноса пяти объектов в соседнюю зону обслуживания. Порядок, положение и ориентация объектов выбирается произвольно);
- транспортировки (сочетание двух первых задач).

При решении используется информация о начальных и конечных положениях объектов, подлежащих транспортировке. Порядок выполнения задач и конкретные действия должны определяться роботом автономно. После размещения всех объектов робот должен покинуть арену через выход [2].

Обычно решение такого типа задач требует достаточно длительных подготовительных работ, проведения тренировок и тестирования оборудования. Из-за различных организационных проблем команда BRC была очень ограничена во времени и начала готовиться к соревнованиям лишь за три недели до их открытия.

Устройство ММР. ММР, разработанный командой, содержал следующие основные конструктивные элементы: четырехколесное меканум-шасси* (шасси с омни-колесами†), позволяющее ММР двигаться в любом направлении без изменения ориентации; пятизвенный манипулятор; два лазерных дальнометра-лидара (HOKUYO URG-04LX-UG01 и Rplidar-A3) и камера глубины Intel Realsense.

Шасси ММР (рис. 1, а) состоит из четырех приводов и меканум-колес. Данная кинематическая схема была выбрана для максимальной мобильности робо-

* Колесо Mecanum — роликонесущее колесо, позволяющее роботу двигаться в любом направлении, а также совмещать линейное и вращательное движения. Представляет собой колесо, по всей окружности которого установлены ролики так, что их угол вращения относительно оси колеса составляет 45°. Соответственно, при вращении результирующая сила также направлена под углом. На четырехколесной платформе, комбинируя вращения отдельных колес, можно добиться не только движения вправо-влево, вперед-назад, но и диагонального движения, а также одновременного движения и вращения вокруг собственной оси.

† Омни-колесо представляет собой колесо, на котором смонтировано несколько роликов, установленных на протяжении всей окружности колеса. Угол вращения роликов составляет 45° между ними и осью колеса.

та на плоскости. Манипулятор ММР строился по ангулярной схеме[‡] и имел пять степеней подвижности. Данная конфигурация позволяет ему обеспечить достижение всех точек, лежащих внутри рабочей зоны (рис. 1, б).

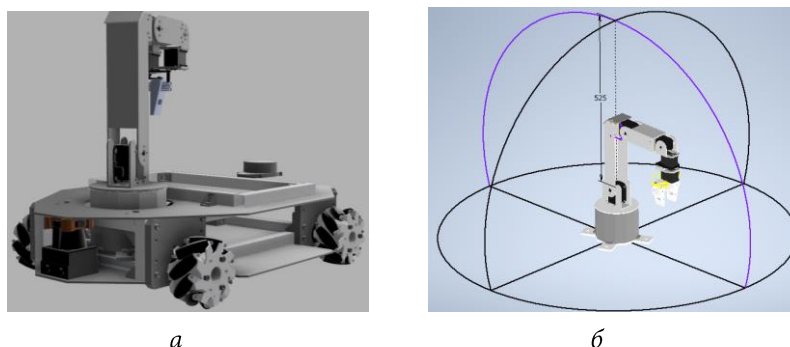


Рис. 1. Модель ММР:

а — внешний вид; б — модель манипулятора в среде Inventor

Для приведения степеней подвижности манипулятора в движение были выбраны сервоприводы Dynamixel, в состав которых входят двигатель постоянного тока, редуктор, датчик угла и микроконтроллер. В качестве интерфейса общения данные сервомоторы используют интерфейс TTL Level Multi Drop, позволяющий соединить устройства в последовательную сеть. Полученная сеть с помощью адаптера USB-DXL-AR подключается к USB порту компьютера управления. В качестве материала корпуса манипулятора применяли полимер в комбинации с алюминиевыми пластинами. В *схвате манипулятора* использовали двухпальцевую схему, включающую маломощный сервомотор Dynamixel, редуктор для преобразования вращательного движения привода в поступательное движение губок и набор крепежных деталей.

Основными элементами *информационной системы* ММР являлись два лазерных дальномера и камера глубины. Первые были использованы для построения карты складского помещения, где ММР перемещался и перевозил объекты, а камера глубины позволяла распознать объект и уточнить координаты его центра тяжести для захвата манипулятором.

Система управления ММР строилась по иерархическому принципу и включала два уровня: верхний, решавший высокоуровневые задачи (локализация робота, планирование его маршрута при движении, распознавание объектов, управление манипулятором и пр.); нижний, управлявший сервоприводами мобильной платформы и проводящий мониторинг заряда аккумулятора.

В качестве вычислителя верхнего уровня использовали мини-ПК Intel NUC с процессором Core i3; для управления приводами манипулятора была разрабо-

[‡] Ангулярная схема — вид схемы манипулятора, при которой он представляет собой незамкнутый контур, каждый элемент которого перемещает определенный привод.

тана специальная преобразовательная плата. На нижнем уровне применяли микроконтроллер ROBOTIS OpenCr под управлением контроллера STM32F746ZGT6 (рис. 2).

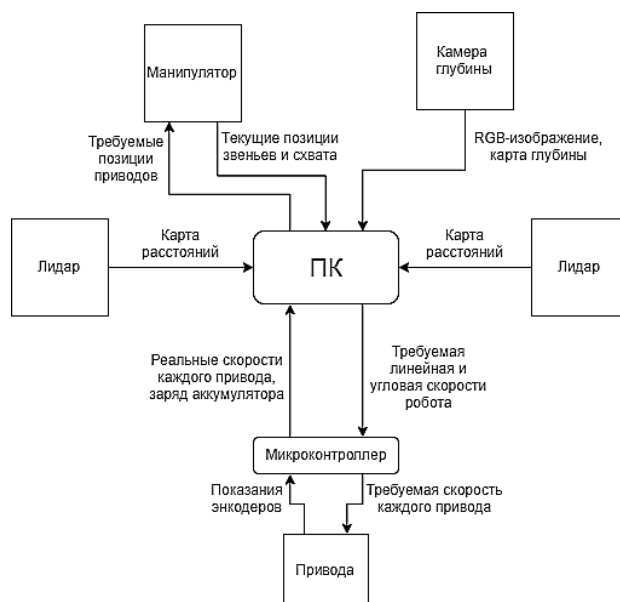


Рис. 2. Структура системы управления ММР

Программное обеспечение ММР. Программное обеспечение ММР строилось на основе архитектуры операционной системы ROS (Robot Operating System) — рис. 3.

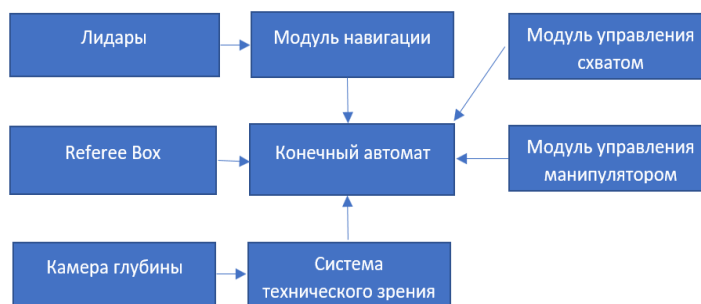


Рис. 3. Основные программные модули ММР

При разработке необходимых алгоритмов с учетом логической структуры заданий и существенных ограничений по вычислительной мощности было принято решение ограничиться формализмом *конечных автоматов*. Программирование каждого конечного автомата проводилось с помощью пакета Smach — стандартного модуля операционной системы ROS, а визуализации его работы — с помощью пакета Smach-viewer.

В качестве примера на рис. 4 представлена схема конечного автомата, обеспечивающего выполнение задачи транспортировки.

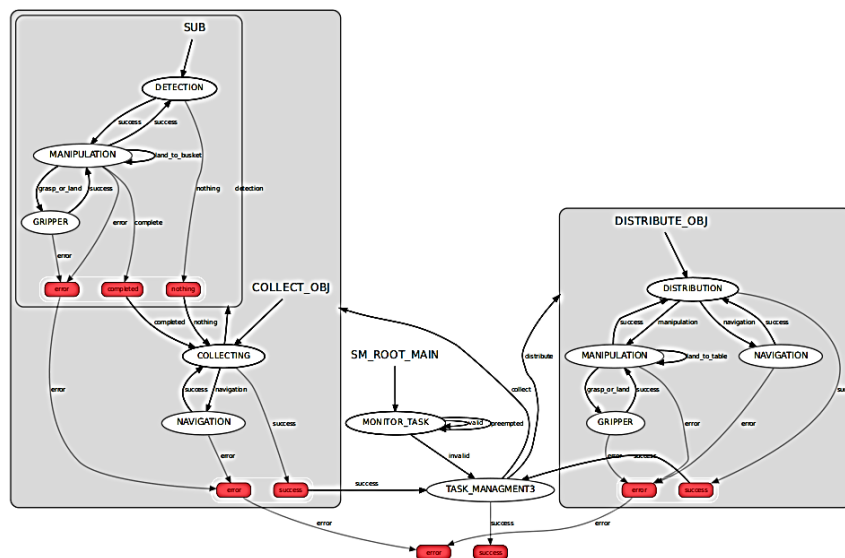


Рис. 4. Конечный автомат решения задачи транспортировки

В соответствии с регламентом соревнований, на отдельном сервере генерируется задание в специальном формате и отправляется роботу по протоколу UDP. Полученное и корректно обработанное задание служит триггером для запуска конечного автомата и начала выполнения задачи. Все условия перехода между состояниями описаны в классах состояния, внутри которых происходит обработка и изменение входных данных. Отметим, что данный автомат оказался достаточно универсальным, что позволило с небольшими изменениями использовать его при реализации алгоритмов навигации и манипулирования объектами.

Шасси ММР. При управлении шасси ММР решались следующие задачи: прямая задача кинематики, состоящая в расчете скорости колес из требуемой линейной и угловой скорости робота; расчет колесной одометрии[§]; построение карты поля и последующая локализация; планирование маршрута с учетом статических препятствий, а также следование по маршруту с учетом динамических препятствий.

Программа навигации ММР была реализована с помощью пакета Navigation Stack операционной системы ROS, который был разбит на несколько модулей: картографирования, локализации и планирования маршрута.

Для картографирования использовали программный пакет Gmapping [3]. На первом этапе при построении карты оператор управляет роботом удаленно. Затем карта редактируется вручную.

[§] Одометрия — использование данных о движении приводов для оценки перемещения.

Далее по построенной карте запускается алгоритм локализации Монте-Карло [4], в основе которого лежит фильтр частиц.

В качестве планировщика пути использовали алгоритм скользящего динамического окна (DWA — Dynamic Window Approach) [5].

Планировщик пути, состоящий из двух функциональных блоков — глобального и локального (рис. 5), является основным элементом навигационной системы; он анализирует положение шасси ММР, определяет текущее состояние карты и показания датчиков. На основе этой информации сначала глобальный планировщик строит маршрут с учетом карты, а затем локальный планировщик перестраивает маршрут в окрестности 2 м от робота с учетом показаний датчиков и отправляет на контроллер шасси необходимые данные о скорости движения.

Структура навигационной системы представлена на рис. 5.

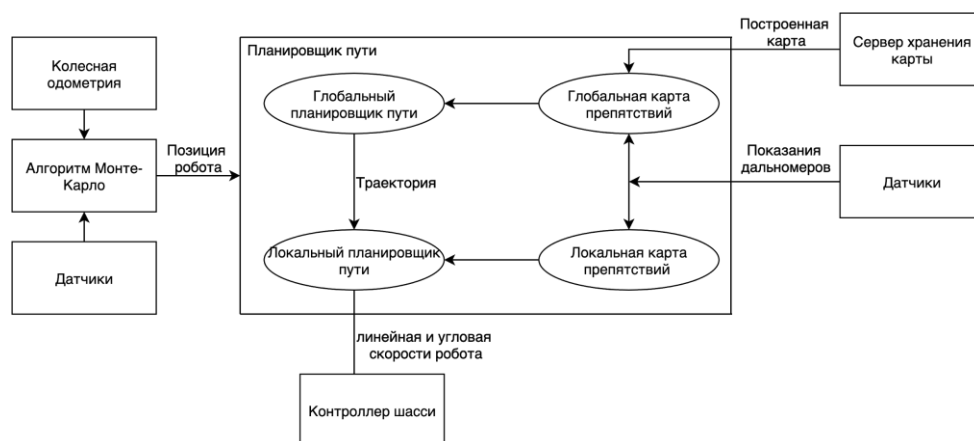


Рис. 5. Структура навигационной системы ММР

Управление манипулятором ММР. Программная часть системы управления манипулятором в процессе разработки была разбита на три функциональных модуля: модуль управления рукой манипулятора, модуль управления приводами и модуль управления схватом (рис. 6). Обработка и передача информации осуществлялись с помощью нод и топиков**.

Модуль управления рукой манипулятора предназначен для решения прямой и обратной задачи о положении. Входным параметром модуля является три координаты точки, где должен располагаться схват манипулятора ММР, а также два угла его ориентации: угол отклонения схвата от горизонтали и угол поворота схвата вокруг собственной оси. Прямую задачу о положении решали известным методом Денавита — Хартенберга [6], обратную — геометрическим мето-

** Ноды и топик — сущности в Robot Operation System, которые позволяют обрабатывать и передавать информацию. Ноды представляют собой программы, которые получают, преобразовывают и отправляют данные к другим нодам. Топики — это каналы передачи информации между нодами.

дом. Данный модуль также сообщает о достижении конечной точки схвата заданного положения.

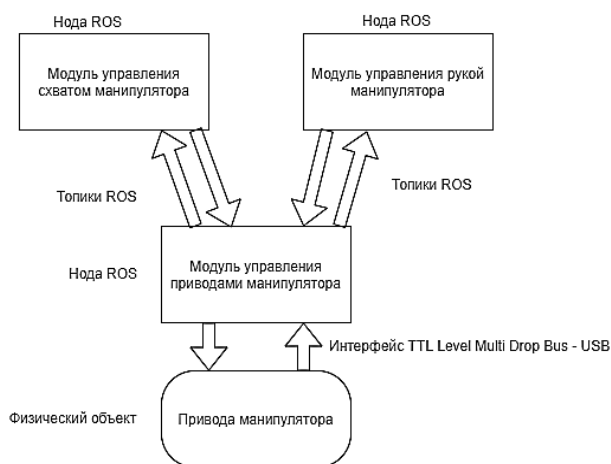


Рис. 6. Схема взаимодействия модулей системы управления манипулятором

Модуль управления приводами манипулятора реализован с помощью готовых библиотек работы с сервомоторами Dynamixel компании Robotis [7]. Благодаря этому появляется возможность задавать желаемые положения приводов, опрашивать энкодеры для получения их текущего положения и т. п. непосредственно через USB-порт персонального компьютера с помощью адаптера TTL Level Multi Drop — USB.

Модуль управления схватом манипулятора принимает на вход команду «открыть/закрыть» схват. Степень сжатия губок схвата ММР определяется с помощью датчика угла поворота привода схвата.

Техническое зрение ММР. Программная реализация системы технического зрения ММР состоит из трех модулей: обнаружения объектов, распознавания объектов и вычисления их ориентации и координат в пространстве (рис. 7). Сначала на изображении с телекамеры детектируются объекты, которые ограничиваются прямоугольными контурами. Затем каждый найденный объект распознается и ему ставится в соответствие определенный класс. После этого для требуемых по заданию объектов вычисляется их положение в пространстве [7].

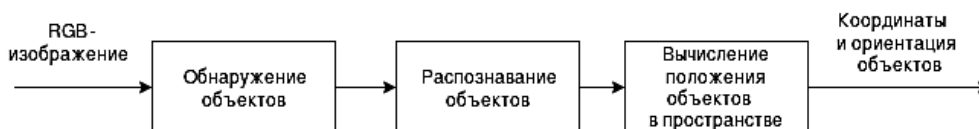


Рис. 7. Структура системы технического зрения

В соответствии с регламентом соревнований необходимо было обнаружить и распознать известные объекты на однотонном фоне. Такая постановка задачи позволила применять классические алгоритмы технического зрения, содержа-

щиеся в открытой библиотеке OpenCV [8]. Последовательность преобразований изображения и некоторые результаты приведены на рис. 8 и 9.

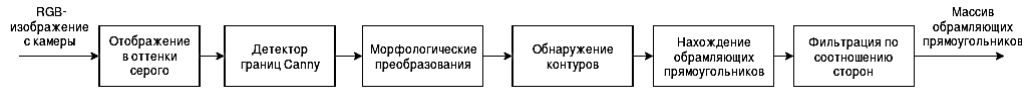


Рис. 8. Структура модуля обнаружения объектов

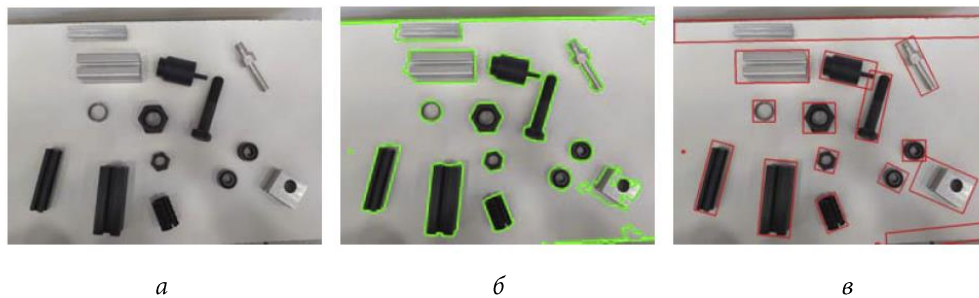


Рис. 9. Результаты работы алгоритмов обработки изображения:

a — исходное изображение; *б* — найденные контуры; *в* — обрамляющие прямоугольники

Дальнейшее распознавание проводили с использованием блока распознавателя, схема которого показана на рис. 10.

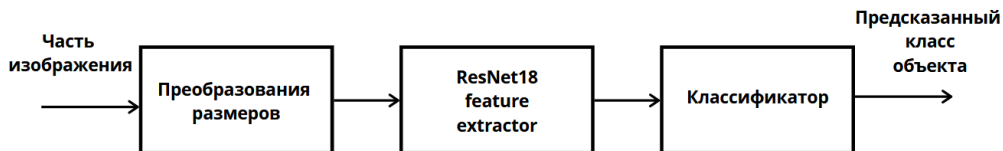


Рис. 10. Схема блока распознавателя

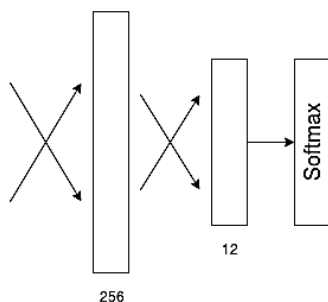


Рис. 11. Структура классификатора

Каждый найденный объект поступал к нему на вход, а на выходе изображению ставился в соответствие определенный класс. Распознаватель состоял из двух основных блоков: детектора особенностей (feature extractor) и классификатора. В качестве детектора особенностей была использована сеть ResNet18 [9], предобученная на специальной выборке изображений ImageNet [10] для распознавания 1000 классов. В рамках соревнований необходимо было распознать 12 классов объектов. Значения с выхода сети подавались на вход классификатора (рис. 11), состоящего из трех слоев: скрытого с 256 нейронами, выходного с

12 и слоя softmax. В скрытом слое использовалась функция активации ReLU и dropout со значением 0,4 [11].

Датасет состоял из 7200 изображений каждого класса (600 на один класс). Он был разбит на три части в соотношении 7/2/1 (обучающая, валидационная и тестовая выборки). Сеть была реализована с использованием фреймворка PyTorch в GoogleColab (рис. 12). Средняя точность на тестовых данных составила порядка 85 %, скорость распознавания одного объекта — 0,20...0,02 с.

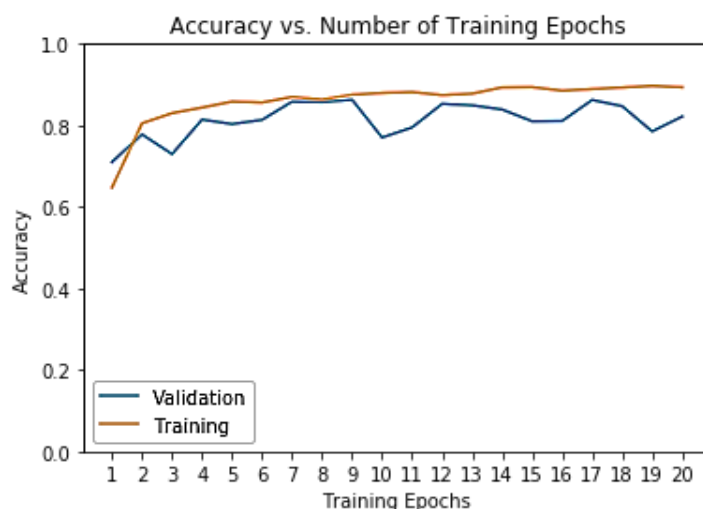


Рис. 12. Кривая изменения точности сети во время обучения (по горизонтали — номер эпохи, по вертикали — значение точности, оранжевая кривая — точность на обучающих данных, синяя — на валидационных)

После определения координат требуемого объекта в пикселях изображения вычисляются реальные координаты объекта относительно камеры в миллиметрах. При сопоставлении данных с изображения и с карты глубины для получения координат объектов используются тригонометрические преобразования

$$\theta = \frac{\pi - \alpha}{2} + dx \frac{\alpha}{w},$$

$$x = \frac{dz}{\operatorname{tg} \theta},$$

где θ — угол между центральной осью камеры и вектором, проведенным от пикселя, отвечающего за распознаваемый объект к самому объекту; α — угол обзора камеры вдоль оси X ; dx — разность между пикселем, показывающим центр распознанного объекта и пикселем центра изображения по оси X ; w — ширина изображения в пикселях; dz — расстояние от линии камеры до объекта, получаемое из карты глубины; x — координата объекта относительно камеры

Аналогичным способом вычисляется координата по оси y :

$$\gamma = 2\pi - \frac{\beta}{2} + dy \frac{\beta}{h};$$
$$y = dz \operatorname{tg} \gamma,$$

где β — угол обзора камеры вдоль оси Y ; dy — разность между пикселем, показывающим центр распознанного объекта и пикселем центра изображения по оси Y ; h — высота изображения в пикселях; y — координата объекта относительно камеры.

Заключение. При подготовке к соревнованиям командой BRC в кратчайшие сроки был спроектирован и изготовлен ММР на основе четырехколесного шасси с омни-колесами и пятизвенного ангулярного манипулятора (рис. 13).

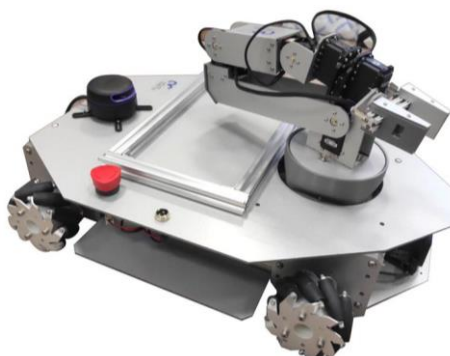


Рис. 13. Финальная модель ММР

Тестовые заезды в рамках RoboCup@Work выявили ряд недостатков конструкции и программного обеспечения робота, которые приходилось устранять непосредственно в процессе соревнований. Были изменены положения лидаров, модифицирована система питания и др. Изменения в программное обеспечение вносили даже непосредственно в процессе движения ММР. Работа над проектом продолжается, сейчас идет процесс переосмысления конструкции и отбор подходящих конструктивных решений. Также разрабатывается новая система навигации и исследуются возможности использования графических вычислителей для увеличения качества распознавания объектов и более точного определения их координат.

Литература

- [1] Carstensen T., Carstensen J., Dick A., et al. Staying on top at RoboCup@Work 2016. *RoboCup 2016: Robot World Cup XX*. Springer, 2016, pp. 601–612.
- [2] Norouzi A., Schnieders B., Zug S., et al. RoboCup@Work rulebook. Robocup, 2019
- [3] Thrun S., Burgard W., Fox D. Probabilistic robotics. The MIT Press, 2005.
- [4] Dellaert F., Fox. D., Burgard W., et al. Monte Carlo localization for mobile robots. *Proc. IEEE Int. Conf. Robot. Automat.*, 1999. DOI: <https://doi.org/10.1109/ROBOT.1999.772544>

- [5] Fox D., Burgard W., Thrun S. The dynamic window approach to collision avoidance. *IEEE Robot. Autom. Mag.*, 1997, vol. 4, no. 1, pp. 23–33. DOI: <https://doi.org/10.1109/100.580977>
- [6] Dynamixel workbench. 1. Introduction. *emanual.robotis.com: веб-сайт*. URL: http://emanual.robotis.com/docs/en/software/dynamixel/dynamixel_workbench/ (дата обращения: 15.09.19).
- [7] Зенкевич С.Л., Ющенко А.С., ред. Основы управления манипуляционными роботами. М., Изд-во МГТУ им. Н.Э. Баумана, 2004.
- [8] Redmon J., Divvala S., Girshick R., et al. You only look once: unified, real-time object detection. *IEEE CVPR*, 2016. DOI: <https://doi.org/10.1109/CVPR.2016.91>
- [9] Deng J., Dong W., Socher R., et al. ImageNet: a large-scale hierarchical image database. *IEEE CVPR*, 2009. DOI: <https://doi.org/10.1109/CVPR.2009.5206848>
- [10] Canny J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1986, vol. 8, no. 6, pp. 679–698. DOI: <https://doi.org/10.1109/TPAMI.1986.4767851>
- [11] He K., Zhang X., Ren Sh., et al. Deep residual learning for image recognition. *IEEE CVPR*, 2016. DOI: <https://doi.org/10.1109/CVPR.2016.90>

Анисимов Родион Олегович — студент кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Бакаев Владислав Сергеевич — студент кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Бахов Тамерлан Беспанович — студент кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Голобурдин Николай Владимирович — студент кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Марчук Александр Михайлович — студент кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Мостаков Николай Алексеевич — студент кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Научный руководитель — Воротников Сергей Анатольевич, кандидат технических наук, доцент кафедры «Робототехнические системы и мехатроника», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Ссылку на эту статью просим оформлять следующим образом:

Анисимов Р.О., Бакаев В.С., Бахов Т.Б., Голобурдин Н.В., Марчук А.М., Мостаков Н.А. Мобильный манипуляционный робот для международных соревнований по робототехнике. *Политехнический молодежный журнал*, 2020, № 11(52). <http://dx.doi.org/10.18698/2541-8009-2020-11-656>

MOBILE MANIPULATION ROBOT FOR INTERNATIONAL ROBOTICS COMPETITION

| | |
|------------------------|---|
| R.O. Anisimov | rodion_anisimov@mail.ru SPIN-code: 2973-6260 |
| V.S. Bakaev | bakaev.98@bk.ru SPIN-code: 4166-4544 |
| T.B. Bakhov | tamelan.baxov@mail.ru SPIN-code: 7933-2300 |
| N.V. Goloburdin | nickolaygoloburdin@mail.ru SPIN-code: 4454-8951 |
| A.M. Marchuk | Marchuk.sasha2010@yandex.ru SPIN-code: 9253-5492 |
| N.A. Mostakov | nikrus333@gmail.com SPIN-code: 5393-7773 |

Bauman Moscow State Technical University, Moscow, Russian Federation

Abstract

The article is devoted to the experience of the participation of BMSTU students team in the RoboCup robotics championship in the RoboCup@Work nomination. This championship is one of the most prestigious student events in which universities from all over the world take part. This nomination is aimed at imitating the actions of a robot in a warehouse. The main tasks lie in the area of navigation and object manipulation. A description of the main subsystems of a mobile manipulation robot is given: a navigation system, a vision system based on a combination of deep learning algorithms and classical methods of image processing, a manipulation system and a finite state machine.

Keywords

Mobile manipulation robot, RoboCup, RoboCup@Work, Bauman Robotics Club, state machine, navigation, technical vision, RealSense, manipulator

Received 09.11.2020

© Bauman Moscow State Technical University, 2020

References

- [1] Carstensen T., Carstensen J., Dick A., et al. Staying on top at RoboCup@Work 2016. *RoboCup 2016: Robot World Cup XX*. Springer, 2016, pp. 601–612.
- [2] Norouzi A., Schnieders B., Zug S., et al. RoboCup@Work rulebook. Robocup, 2019
- [3] Thrun S., Burgard W., Fox D. Probabilistic robotics. The MIT Press, 2005.
- [4] Dellaert F., Fox D., Burgard W., et al. Monte Carlo localization for mobile robots. *Proc. IEEE Int. Conf. Robot. Automat.*, 1999. DOI: <https://doi.org/10.1109/ROBOT.1999.772544>
- [5] Fox D., Burgard W., Thrun S. The dynamic window approach to collision avoidance. *IEEE Robot. Autom. Mag.*, 1997, vol. 4, no. 1, pp. 23–33. DOI: <https://doi.org/10.1109/100.580977>
- [6] Dynamixel workbench. 1. Introduction. *emanual.robotis.com: website*. URL: http://emanual.robotis.com/docs/en/software/dynamixel/dynamixel_workbench/ (accessed: 15.09.19).

- [7] Zenkevich S.L., Yushchenko A.S., eds. *Osnovy upravleniya manipulyatsionnymi robotami* [Control basics of manipulation robots]. Moscow, Bauman MSTU Publ., 2004 (in Russ.).
- [8] Redmon J., Divvala S., Girshick R., et al. You only look once: unified, real-time object detection. *IEEE CVPR*, 2016. DOI: <https://doi.org/10.1109/CVPR.2016.91>
- [9] Deng J., Dong W., Socher R., et al. ImageNet: a large-scale hierarchical image database. *IEEE CVPR*, 2009. DOI: <https://doi.org/10.1109/CVPR.2009.5206848>
- [10] Canny J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1986, vol. 8, no. 6, pp. 679–698. DOI: <https://doi.org/10.1109/TPAMI.1986.4767851>
- [11] He K., Zhang X., Ren Sh., et al. Deep residual learning for image recognition. *IEEE CVPR*, 2016. DOI: <https://doi.org/10.1109/CVPR.2016.90>

Anisimov R.O. — Student, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

Bakaev V.S. — Student, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

Bakhov T.B. — Student, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

Goloburdin N.V. — Student, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

Marchuk A.M. — Student, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

Mostakov N.A. — Student, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

Scientific advisor — Vorotnikov Sergey Anatolievich, Cand. Sc. (Eng.), Assoc. Professor, Department of Robotics and Mechatronics, Bauman Moscow State Technical University, Moscow, Russian Federation.

Please cite this article in English as:

Anisimov R.O., Bakaev V.S., Bakhov T.B., Goloburdin N.V., Marchuk A.M., Mostakov N.A. Mobile manipulation robot for international robotics competition. *Politekhnicheskii molodezhnyy zhurnal* [Politechnical student journal], 2020, no. 11(52). <http://dx.doi.org/10.18698/2541-8009-2020-11-656.html> (in Russ.).