

18. Pardo-Castellote G. Omg data-distribution service: Architectural overview, *23rd International Conference on Distributed Computing Systems Workshops, 2003. Proceedings*. IEEE, 2003, pp. 200-206.
19. Voronov N.V., Gimpel'son V.D., Maslov M.V. [et al.]. Sistema dinamicheskoy dvoichnoy translyatsii x86 → «El'brus» [Dynamic binary translation system x86 → "Elbrus"], *Voprosy radioelektroniki* [Questions of radio electronics], 2012, Vol. 4, No. 3, pp. 89-108. EDN OUMJOF.
20. Operatsionnye sistemy «El'brus» [Elbrus operating systems]. Available at: http://www.mcst.ru/Elbrus_OS (accessed 25 February 2026).
21. Bocharov N.A. Issledovanie podkhodov k unifikatsii bortovykh vychislitel'nykh kompleksov [Study of approaches to the unification of onboard computing systems], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2023, No. 1 (231), pp. 275-287. DOI 10.18522/2311-3103-2023-1-275-287. – EDN AIYGIE.
22. Rukovodstvo po effektivnomu programmirovaniyu na platforme «El'brus» [Guide to effective programming on the Elbrus platform]. Available at: http://mcst.ru/doc/elbrus_prog/elbrus-prog-1.2_2024-02-28.pdf (accessed 25 February 2026).

Бочаров Никита Алексеевич – ПАО «ИНЭУМ им. И.С. Брука»; e-mail: bocharov.na@phystech.edu; г. Москва, Россия; тел.: +79167346437; д.т.н.; зам. руководителя управления – главный научный сотрудник.

Суминов Константин Александрович – ПАО «ИНЭУМ им. И.С. Брука»; e-mail: suminov_k@ineum.ru; 119334, г. Москва, Россия; тел.: +74991355336; к.т.н.; начальник отдела.

Кириллук Михаил Андреевич – ПАО «ИНЭУМ им. И.С. Брука»; e-mail: kirilyuk_m@ineum.ru; г. Москва, Россия; тел.: +79055273011; к.т.н.; начальник отдела.

Парамонов Николай Борисович – ПАО «ИНЭУМ им. И.С. Брука»; e-mail: paramon_n@ineum.ru; г. Москва, Россия; тел.: +74991355336; д.т.н.; профессор; руководитель управления.

Bocharov Nikita Alekseevich – JSC «INEUM»; e-mail: bocharov.na@phystech.edu; Moscow, Russia; phone: +79167346437; dr. of eng. sc.; deputy head of division – chief researcher.

Suminov Konstantin Aleksandrovich – JSC «INEUM»; e-mail: suminov_k@ineum.ru; Moscow, Russia; phone: +74991355336; cand. of eng. sc.; head of department.

Kirilyuk Mikhail Andreevich – JSC «INEUM»; e-mail: kirilyuk_m@ineum.ru; Moscow, Russia; phone: +79055273011; cand. of eng. sc.; head of department.

Paramonov Nikolay Borisovich – JSC «INEUM»; e-mail: paramon_n@ineum.ru; Moscow, Russia; phone: +74991355336; dr. of eng. sc.; professor; head of division.

УДК 004.853

DOI 10.18522/2311-3103-2026-2-271-285

И.С. Фомин, А.А. Базельцев

ИСПОЛЬЗОВАНИЕ СИСТЕМЫ КАМЕР КРУГОВОГО ОБЗОРА ДЛЯ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ ДВИЖЕНИЯ МОБИЛЬНОГО РОБОТА

Рассматривается проблема обеспечения безопасности движения мобильной робототехнической платформы в окружении с известными классами объектов на основе результатов обнаружения объектов по системе камер кругового обзора. Хорошо известны решения по обеспечению безопасности по камерам с использованием оптических потоков или иных способов выделения подвижных объектов в поле зрения. Здесь предлагается использовать для обеспечения безопасности положение и форму объектов, рассчитанные по результатам работы нейросетевого детектора. Система кругового обзора (СКО) состоит из 4 камер с широким углом зрения. Рассмотрены несколько подходов к подаче объектов в нейросетевой детектор. Показаны различия по качеству и быстродействию для этих подходов, сформированы методические рекомендации по использованию. Для расчета положений объектов в системе координат камер и через положения камер – в СК робота – предложены 2 решения на основе калибровки камер и некоторых допущений. В первом случае положение робота на поверхности принимается за горизонтальное, а высота камеры над поверхностью – постоянной и известной, а во втором – предварительно известным принимается один метрический размер объекта для каждого класса. Предложенное решение

развернуто и протестировано на вычислителе на базе Rockchip 3588 и показывает высокие результаты по количеству обнаружений (от 8 до 19 объектов в среднем на кадр в зависимости от настроек), и быстродействию (0,29 с для 8 объектов на кадр и около 1,05 с для результата в 19 объектов на кадр). С точки зрения точности определения дистанции до объектов, то для первого способа стандартное отклонение составило от 4,2 до 7,9 мм, для второго - от 3,8 до 7,6 мм. Устойчивость оценки размеров объектов для первого способа от 0,64 до 2,02 мм, для второго от 1,63 до 2,22 мм соответственно. Полученные результаты позволяют уверенно утверждать, что предложенный алгоритм обнаружения объектов применим для обеспечения безопасности и навигации мобильного робота с использованием камер системы кругового обзора.

Обнаружение объектов; обеспечение безопасности; система кругового обзора; встраиваемый вычислитель; YOLO.

I.S. Fomin, A.A. Baseltsev

APPLICATION OF A SURROUND-VIEW CAMERA SYSTEM FOR MOBILE ROBOT MOVEMENT SAFETY

This paper addresses the problem of ensuring the motion safety of a mobile robotic platform in an environment with known object classes, based on object detection results from a surround-view camera system. While solutions based on optical flow or other methods for detecting moving objects in a camera's field of view are well-known, this work proposes to use the results from a dedicated object detector for safety assurance. Here, we propose to utilize object positions, calculated from the output of a neural network detector, for safety purposes. The surround-view camera system (SVS) consists of 4 cameras with wide-angle lenses. Several approaches for feeding objects into the neural network detector are considered. Differences in quality and performance for these approaches are demonstrated, and methodological recommendations for their use are formulated. To calculate object positions in the camera coordinate system and, through camera positions, in the robot's coordinate frame, 2 solutions based on camera calibration and certain assumptions are proposed. In the first case, the robot's position on the surface is assumed to be horizontal, and the camera's height above the surface is assumed constant and known. In the second case, one of the metric dimensions of objects for each class are assumed to be known in advance. The proposed solution has been deployed and tested on a Rockchip 3588-based computing platform, demonstrating high performance in terms of detection count (from 8 to 19 objects per frame on average, depending on settings) and processing speed (0.29 s for 8 objects per frame and about 1.05 s for the result of 19 objects per frame). Regarding the accuracy of distance estimation to objects, for the first method, the standard deviation ranged from 4.2 to 7.9 mm, and for the second method, from 3.8 to 7.6 mm. The standard deviation of object size estimation for the first method ranged from 0.64 to 2.02 mm, and for the second method from 1.63 to 2.22 mm, respectively. The obtained results allow us to confidently state that the proposed object detection algorithm is applicable for ensuring the safety and navigation of a mobile robot using surround-view system cameras.

Object detection; movement safety; surround-view camera system; embedded computing platform; YOLO.

Введение. Задача обнаружения объектов и определения их положения в окружении мобильного транспортного средства является хорошо известной и давно решаемой в задаче обеспечения безопасности при автовождении. В частности, есть решения на основе оптических потоков, которые предлагались в том числе и в ЦНИИ РТК [1]. Основной способ решения задачи обеспечения безопасности крупных транспортных средств сейчас – использование одного или нескольких 3D LiDAR для оценки препятствий по всем угрозам направлениям [2]. Это хорошо работающий подход, но он требует существенных вычислительных затрат, которые можно легко позволить себе при разработке крупногабаритного или среднегабаритного мобильного робота или транспортного средства [3–5], где допустим мощный вычислитель с высоким энергопотреблением или обмен данными для обработки на серверах.

В наземной сервисной мобильной робототехнике крайне важны автономность и низкое энергопотребление, поэтому решение задачи обеспечения безопасности для них предполагает использование сенсоров и вычислителей с низким энергопотреблением, осуществление всех расчетов на борту мобильного робота и наличие минимальных априорных знаний о том окружении, в котором роботу предстоит работать. Так как для

управления оператором мобильный робот уже оснащается системой камер кругового обзора, разумно использовать ее для автоматического обнаружения препятствий и оценки расстояния до них с целью обеспечения безопасности перемещения.

Важной особенностью является использование камер с широким или сверхшироким углом обзора, что приводит к существенным дисторсионным искажениям изображения. Это требует применения специальных алгоритмов калибровки камеры [6], обработки результатов обнаружения для компенсации дисторсионных искажений и построения особых алгоритмов расчета расстояния и оценки формы объектов с учетом того, что с достаточной точностью можно измерить только угол направления на объект относительно вектора оптической оси.

Постановка задачи. Основной целью данной работы является задача обнаружения потенциально опасных объектов при автономном движении малоразмерного мобильного робота, оснащенного системой широкоугольных камер для обеспечения кругового обзора и маломощным встраиваемым вычислителем. Для обеспечения безопасности необходимо решить задачи обнаружения объектов с достаточными быстродействием, точностью и полнотой, а также устойчивого вычисления расстояния до объектов и определения их размеров по камерам с сильными дисторсионными искажениями в монокулярной постановке.

На решение задачи обнаружения объектов накладывается ряд существенных ограничений. Для расчета нейронных сетей наилучшим решением являются встраиваемые вычислители на базе Nvidia, но из-за недоступности к свободному приобретению их использование невозможно. Выбран вычислитель Orange Pi 5 Ultra с встроенным модулем тензорных вычислений TPU, построенный на базе процессора Rockchip 3588. Для использования на данном вычислителе требуется перевод нейронной сети в обобщенный формат ONNX и последующая компиляция весов в специальный формат RKNN. Это накладывает существенные ограничения на доступный набор моделей (общим местом является применение сетей семейства YOLO [7], обученных на больших наборах данных [8]). Требуется провести анализ существующих моделей по соотношению качества и быстродействия и выбрать наиболее подходящую с учетом ограничений вычислителя. Доступный объем памяти ограничивает размер входа модели, поэтому необходимо разработать подход к обработке, оптимальный с точки зрения соотношения быстродействия, точности и полноты обнаружений.

Для решения задачи вычисления расстояния до объектов и оценки их размеров требуется разработать специальные алгоритмы. Традиционно используются подходы на базе стереокамер или камер глубины с неширокими углами обзора. В силу использования монокулярной камеры с широким углом обзора требуется разработать специфические алгоритмы расчета дистанции на основе доступной информации. В работе предлагается два алгоритма, опирающиеся на параметры внутренней калибровки, априорную информацию о геометрии робота и его расположении относительно поверхности в первом случае и априорные данные о характерных геометрических размерах для определенных классов объектов во втором случае.

Существующие системы кругового обзора и алгоритмы калибровки широкоугольных камер. Так как для оператора будет в том числе формироваться панорамное изображение, конфигурация и параметры камер должны это учитывать. Рассмотрим существующие подходы к формированию панорам.

Первый известный подход основан на использовании одной сверхширокоугольной камеры, направленной вертикально вверх [9], он широко применяется в системах наблюдения или роботах-пылесосах. Главным недостатком такой СКО является небольшой угол обзора по вертикали и полное отсутствие зон стереоскопии при использовании моноскопической камеры, из-за этого – невозможность получения истинной глубины. Есть модификации с использованием отражающего элемента, также компактные и имеющие больший угол обзора по вертикали [10], они используются в системах автовождения [11] и на мобильных роботах [12]. К существенным недостаткам данного варианта системы можно отнести использование зеркала: камера при ярком источнике света с одной стороны робота получает блики на весь кадр, в связи с чем не происходит фиксации пригодного

го к анализу изображения. Второй подход к формированию панорамы на основе одной камеры – использование монокулярной камеры с поворотным механизмом. Для получения данных роботу с такой системой необходимо останавливаться через небольшие промежутки времени, получать панораму окружения и проводить сопоставление кадров. Такой алгоритм сильно снижает эффективность системы безопасности и допустимую скорость робота, а также повышает потребление питания. Пример использования такой системы – камеры WAC марсохода EхoMars [13].

Второй способ формирования панорамного изображения основан на использовании системы из нескольких камер. С одной стороны, это позволяет существенно улучшить обзор препятствий вокруг робота и повысить качество получаемых изображений, а с другой – вносит дополнительные сложности с синхронизацией видеопотоков с различных камер, особенно при построении оценок положений объектов в движении и при поворотах. Примеры таких панорамных систем можно увидеть в работах [14], или [15], где рассматриваются проблемы сшивки объектов на разных дистанциях с учетом параллакса. Важный недостаток таких систем – возрастание энергопотребления с увеличением количества камер, и кратный рост вычислений, которые требуются для обработки нескольких кадров. Частично борьба с ним посвящена и данная работа.

Необходимо отметить также существование сложных систем кругового обзора, которые используют несколько камер различного типа, в том числе с зеркалами [16], пусть они и не являются предметом рассмотрения данной работы.

Как для сшивки панорамного изображения, так и для вычисления положения объекта в поле зрения по одной камере требуется решение задачи калибровки камеры широкого угла обзора и взаимной юстировки камер системы кругового обзора. Методы калибровки широкоугольных и сверхширокоугольных камер можно условно разделить на методы «самокалибровки» и фотограмметрической калибровки. При самокалибровке предполагается оценка внутренних параметров камеры на основе нескольких кадров, полученных путем перемещения камеры в статической сцене [17], однако такие подходы требуют сложных вычислений и выполнения многих условий, а также плохо применимы для калибровки многокамерных систем. Обычно используется фотограмметрическая калибровка, основанная на получении и анализе изображений с различными калибровочными паттернами. На основании большого количества снимков одного и того же паттерна, расположенного в разных частях кадра под разным углом к статической камере, можно получить связь между 2D картинкой и реальным миром. Паттерны всегда имеют известную геометрию и бывают различного типа. Широкое распространение получили паттерны на основе легко детектируемых примитивов. Известны паттерны типа «шахматная доска» [18] и круговой паттерн [19]. Процедура калибровки включает в себя получение кадров с паттерном, обнаружение особых точек и расчет их пиксельных координат, получение начального приближения калибровки и вычисление параметров камеры как результата решения системы уравнений посредством минимизации фотограмметрической ошибки. Различия в методах калибровки состоят в основном в используемых уравнениях. Для камер с широким углом зрения общепринятыми являются модели калибровки Скарамужза (D. Scaramuzza) [6] и Канналы-Брандта (J. Kannala, S.S. Brandt) [20]. Модель Скарамужза предназначена для калибровки только широкоугольных и сверхширокоугольных камер. Отличительной особенностью модели является использование разложенного полинома в ряд Тейлора для получения коэффициентов внутренних параметров. Модель используется для сложных систем, например, катадиоптрических камер или камер с углом обзора больше 180°. Данный метод реализован в программной среде Matlab. Модель Канналы-Брандта является универсальной и применяется для калибровки камер с различным углом обзора. Отличительной чертой модели является использование радиально-симметричного полинома для вычисления внутренних параметров. Данный метод требует достаточно большого количества кадров для калибровки. Модель используется в библиотеке OpenCV для калибровки широкоугольных камер. В работе для получения параметров камер использовалась калибровка фотограмметрическим методом с использованием паттерна шахматной доски и модель Скарамужза.

Обзор алгоритмов обнаружения объектов. Также необходимо дать краткую характеристику существующих алгоритмов обнаружения объектов. Для обнаружения объектов на изображениях используются одноэтапные и двухэтапные нейросетевые детекторы. Выбор подхода зависит от требований к скорости, точности и устойчивости алгоритма к различным условиям, включая наличие мелких объектов, сложный фон и др.

Одноэтапные детекторы выполняют обнаружение за один прямой проход через сеть, что обеспечивает высокую скорость и делает их подходящими для задач обнаружения в реальном времени. Например, SSD (Single Shot Detector) [21] предсказывает объекты на разных масштабах карты признаков, что ускоряет работу, но снижает точность при наличии объектов малого размера. CenterNet [22] вместо прямого предсказания использует тепловые карты, где каждая точка обозначает центр объекта, а его размеры и смещения уточняются отдельно – это повышает точность и устойчивость к шуму. RetinaNet [23] справляется с несбалансированностью классов благодаря фокальной функции потерь (focal loss), снижая влияние фоновых областей и повышая качество предсказаний. EfficientDet [24] использует легковесную архитектуру EfficientNet и эффективную схему объединения признаков (BiFPN), обеспечивая хорошее соотношение точности и скорости. YOLO (You Only Look Once) [25] проводит одновременное предсказание координат и классов объектов на изображении, благодаря чему остаётся одним из самых быстрых и популярных решений для детекции в реальном времени.

Двухэтапные детекторы сначала формируют регионы-кандидаты (region proposals), а затем классифицируют и уточняют их, что повышает точность, особенно при обнаружении мелких, перекрытых или слабо выраженных объектов. Faster R-CNN [26] – классический пример такого подхода, в котором Region Proposal Network выделяет потенциальные области, после чего отдельная сеть классифицирует содержимое и уточняет границы. Это обеспечивает высокую точность локализации, но снижает скорость обработки. Mask R-CNN [27] дополняет Faster R-CNN дополнительным выходом для предсказания маски сегментации, позволяя выделять точную форму объекта, что особенно полезно в медицинских или промышленных задачах. Более современный Sparse R-CNN [28] отказывается от полного перебора регионов и использует ограниченное количество обучаемых предположений, которые адаптируются в процессе обучения, повышая производительность модели.

Использованная модель сети обнаружения объектов. Большинство представленных моделей отличаются тем, что используют в той или иной степени нестандартные слои, оригинальные способы формирования функции ошибки и т.д., откуда следует обычно невозможность их использования на вычислителях, которые не относятся к семейству Nvidia. Несмотря на достаточно широкую номенклатуру строительных единиц архитектур в формате ONNX [29], большинство авторов не прилагают усилий по адаптации архитектур к переносу в этот формат, и далее, через него в аппаратно ускоряемые форматы. Фактически из всех рассмотренных выше архитектур перенос на вычислитель RK3588 допускают только модели семейства YOLO. Именно модель этого семейства YOLOv8m была выбрана для экспериментов. Что касается более современных версий YOLO – v11, v12 [30] – они были проверены, но не были использованы, т.к. для них на момент проведения экспериментов отсутствовали веса, обученные на OpenImages, а стандартные веса на 80 классов не отвечали требованиям задачи (принципиально не обнаруживали часть требуемых по условиям объектов). Практическое сравнение качества на видео с используемых в экспериментах камер не показало явного преимущества старших версий перед v8m, поэтому были использованы веса YOLOv8m, обученные на наборе данных OpenImages v7 [8] и преобразованные в формат RK3588 с использованием фреймворка Ultralytics [31]. Для запуска модели непосредственно на борту вычислителя использовался фреймворк ezrknn-toolkit2 [32].

Разработанные подходы к обработке данных нескольких камер. Обработка данных с нескольких камер кратно увеличивает время обнаружения объектов по сравнению с одной камерой. Для обработки данных с системы кругового обзора предложено несколько различных подходов. Общее для всех подходов – захват с камер выполняется в

отдельных узлах ROS2, после чего к изображениям можно обращаться в других узлах через топики изображений. В узле обработки данных с камер захват изображений из топиков осуществляется в асинхронном режиме с искусственной многопоточностью. Полученные изображения помещаются в переменные для обработки через блокировку. Размер очереди в 1 элемент позволяет получать самое актуальное изображение сразу, как только завершена работа по извлечению предыдущего. Изображения, поступившие во время извлечения, пропускаются. При обработке одновременно извлекаются и обрабатываются самые актуальные изображения на момент начала итерации работы потока обработки. Все это в совокупности позволяет получить максимально синхронизированные и актуальные данные на момент обработки.

При обработке данных с камер СКО были рассмотрены несколько подходов. Результаты их сравнения показаны в табл. 1. Самый простой – последовательный вызов нейронной сети, рассчитанной на обработку одного изображения, для каждого из изображений со всех камер. Более сложный подход – объединение 4 изображений с камер в пачку и использование нейронной сети, сконфигурированной для единовременной обработки пачки из 4-х изображений. Несмотря на то, что предполагалось, что такой подход будет быстрее, на практике он показал себя даже медленнее, чем последовательная обработка. Это связано с нехваткой ресурсов у TPU для одновременного вычисления пачки целиком. Также рассматривался вариант с объединением 4-х изображений с камер в одно изображение и подача такого объединенного изображения на нейронную сеть, рассчитанную на обработку одной картинки (по сетке 2×2 , т.е. каждая картинка при обработке вдвое меньше), или на нейронную сеть, рассчитанную на обработку кадра вдвое большего размера (каждое изображение обрабатывается без уменьшения). Вдвое меньшие изображения ожидаемо обрабатываются в 4 раза быстрее, но существенно снижается количество выделяемых объектов. На вдвое большем разрешении падения качества не происходит, но время обработки превышает время покадровой обработки и обработки пачкой. Таким образом, на момент написания статьи основным способом была выбрана покадровая обработка.

Таблица 1

Результаты оценки быстродействия разных вариантов обработки

Тип	Кадр	Средн., с	σ , с	Мин., с	Макс, с	Обнар.
2x2	320p	0,29	0,02	0,2525	0,3828	83
2x2	640p	1,3	0,03	1,2296	1,4047	160
4 пачка	640p	1,1	0,03	1,0397	1,32	190
4 послед.	640p	1,05	0,05	0,9325	1,198	191

Математическая модель камеры Скарамузза. Для качественного вычисления положений объектов в системе координат камеры необходимо получить параметры внутренней калибровки. В данной работе была использована модель камеры Скарамузза [6], как более качественная и устойчивая по сравнению с моделью из библиотеки OpenCV [20], также отмечается, что модель OpenCV становится несостоятельной при поле зрения более 110 градусов по большей стороне, а в используемых камерах поле зрения составляет 160 градусов по паспорту и порядка 140 градусов при эмпирической оценке.

Модель Скарамузза [6], подразумевает наличие двух плоскостей: плоскость изображения камеры (u', v'), совпадающая с плоскостью матрицы камеры (рис. 1,в), и плоскость сенсора (u'', v''), являющаяся гипотетической плоскостью, перпендикулярной оси камеры (рис. 1,б). На рис. 1,а изображены плоскости для катодиоптрической системы, при отсутствии зеркал меняется только знак u'' .

Если взять некоторую точку X на сцене, тогда взаимосвязь этой точки и точки на плоскости сенсора будет характеризоваться некоторым вектором p , направленным от точки обзора O к точке X. Таким образом проекция точки пересечения этого вектора с плоскостью изображения камеры будет отражать взаимосвязь точки на сцене и точки на плоскости сенсора (1).

$$\lambda \cdot p = \lambda \cdot g(u'') = \lambda \cdot g(A \cdot u' + t) = PX, \lambda > 0, \quad (1)$$

где P – матрица перспективной проекции, g – нелинейная функция.

Калибровка камеры подразумевает оценку матриц A и t , а также функции g так, чтобы вектор $g(A \cdot u' + t)$ удовлетворял уравнению, описанному выше.

Функция g имеет вид (2), при этом функция $f(u'', v'')$ является разложением в ряд Тейлора (3).

$$g(u'', v'') = (u'', v'', f(u'', v''))^T. \quad (2)$$

$$f(u'', v'') = a_0 + a_1 \cdot \rho'' + a_2 \cdot \rho''^2 + \dots + a_N \cdot \rho''^N, \quad (3)$$

где $\rho'' = \sqrt{u''^2 + v''^2}$, $a_i, i = 0, 1, 2, \dots, N$ – коэффициенты полинома, N – порядок полинома.

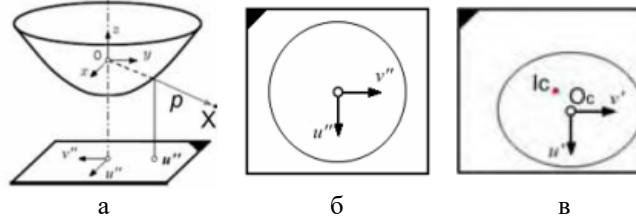


Рис. 1. Модель камеры Скарамuzzi: а – система координат в катадиоптрической системе, б – плоскость сенсора, в – плоскость изображения камеры

Алгоритмы вычисления положения объектов с использованием данных внутренней калибровки. Зная коэффициенты полинома и центр изображения, полученные после калибровки, можно найти некоторые углы в сферической системе координат, которые будут характеризовать луч из центра камеры к точке на сцене. Для начала осуществляется переход в координаты центра изображения (4).

$$\begin{aligned} u' &= u - x_c, \\ v' &= v - y_c, \end{aligned} \quad (4)$$

где x_c, y_c – координаты центра изображения,
 u, v – координаты пикселя в исходном изображении,
 u', v' – координаты относительно центра.

Далее, происходит обратное аффинное преобразование по формуле (5), и получение радиальной величины для заданных координат пикселя (6).

$$\begin{bmatrix} u'' \\ v'' \end{bmatrix} = \begin{bmatrix} c & d \\ e & f \end{bmatrix}^{-1} \begin{bmatrix} u' \\ v' \end{bmatrix}, \quad (5)$$

$$\rho'' = \sqrt{u''^2 + v''^2}, \quad (6)$$

где $\begin{bmatrix} c & d \\ e & f \end{bmatrix}$ – матрица, полученная при калибровке.

Теперь появляется возможность получения координаты луча по оптической оси z , выраженной из полинома (7).

$$z_n = a_0 + a_1 \cdot \rho'' + a_2 \cdot \rho''^2 + \dots + a_N \cdot \rho''^N. \quad (7)$$

Вектор луча в таком случае имеет вид (8). Зная вектор, также можно получить углы в сферической системе координат θ – угол от оси, φ – азимут (9).

$$\bar{d} = \frac{[u'', v'', z_n]^T}{\sqrt{u''^2 + v''^2 + z_n^2}} \quad (8)$$

$$\begin{aligned} \theta &= \cos^{-1} d_z, \\ \varphi &= \tan^{-1} \frac{d_y}{d_x}. \end{aligned} \quad (9)$$

Однако углов или нормированного вектора недостаточно для получения координат изображения, так как они дают только направление к реальной точке, поэтому для координат в реальном мире необходимо знать хотя бы одно реальное значение расстояния на изображении.

Искомый объект находится в плоскости, параллельной плоскости XY , на расстоянии Z_0 , в связи с чем задачу определения расстояния до объекта можно свести к решению задачи пересечения луча с этой плоскостью.

Любая точка на луче выражается формулой (10) для расстояния Z_0 (11), Соответственно координаты точки пересечения луча с плоскостью выражаются формулой (12).

$$P(t) = t \cdot \vec{d}, \quad (10)$$

$$Z_0 = t \cdot d_z, \quad (11)$$

$$t = \frac{Z_0}{d_z},$$

$$P(t) = \frac{Z_0}{d_z} \cdot \vec{d}. \quad (12)$$

И теперь для поиска координат точки достаточно найти Z_0 , помимо этого можно найти координаты до любой точки объекта, так как Z_0 для всех точек постоянна. Найти расстояние Z_0 можно по координате по оси Y_c до точки, по высоте объекта или по ширине объекта.

Определение параметров объекта по известной высоте камеры. Это стандартная ситуация при оценке расстояния до объекта, для которого известно, что он расположен на полу. Тогда центральная нижняя точка рамки объекта используется как опорная при вычислении расстояния, а высота камеры над полом считается известной из конструктива робота. Примем расстояние H от камеры (начала координат) до искомой точки по оси Y_c известным. Искомая точка представлена на рис. 2. Тогда, подставив известную координату в уравнение точки, можно выразить Z_0 (13).

$$H = \frac{Z_0}{d_z} \cdot d_y, \quad (13)$$

$$Z_0 = \frac{d_z}{d_y} \cdot H.$$

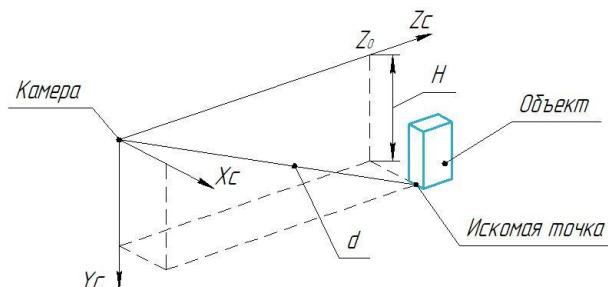


Рис. 2. Определение расстояния от камеры до точки с известной высотой

Используя рассчитанное расстояние до плоскости, перпендикулярной оптической оси камеры, и отклонение объекта от центральной точки несложно определить координаты объекта в проекции на поверхность перемещения (14).

$$Y = H, X = \frac{Z_0}{d_z} \cdot d_x. \quad (14)$$

Определение параметров объекта по известным метрическим размерам. Так как нейронная сеть позволяет в числе прочего оценить класс объекта, по известному классу для отдельных категорий объектов можно предварительно оценить их геометрические размеры, например, столы, стулья, мониторы, системные блоки ПК – имеют не сильно отличающиеся геометрические размеры, и в отсутствие иных опорных данных дистанция до объекта и его положение относительно камеры могут быть оценены с некоторой погрешностью по известным параметрам внутренней калибровки.

В случае известной высоты объекта в реальности в метрах и на изображении в пикселях, можно оценить расстояние до объекта с помощью разности координат верхней точки и нижней точки объекта, которые представлены на рис. 3.

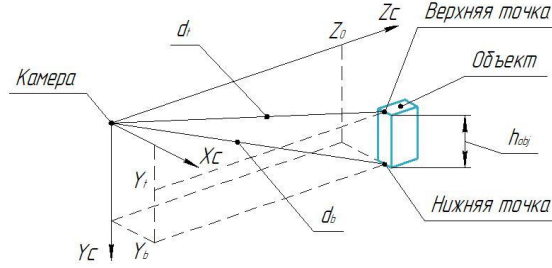


Рис. 3. Вычисление расстояния по известным верхней и нижней точке

Координата верхней точки принимается равной Y_t , нижней точки Y_b , тогда высота объекта h_{obj} вычисляется по формуле (15), на плоскости эти точки выражаются как (16).

$$h_{obj} = Y_b - Y_t, \quad (15)$$

$$Y_b = \frac{Z_0}{d_{zb}} \cdot d_{yb}, \quad (16)$$

$$Y_t = \frac{Z_0}{d_{zt}} \cdot d_{yt}.$$

Соответственно, высота объекта может быть выражена через формулы (17), и уже из данных соотношений можно выразить Z_0 (18).

$$h_{obj} = \frac{Z_0}{d_{zb}} \cdot d_{yb} - \frac{Z_0}{d_{zt}} \cdot d_{yt}, \quad (17)$$

$$h_{obj} = Z_0 \cdot \left(\frac{d_{yb}}{d_{zb}} - \frac{d_{yt}}{d_{zt}} \right),$$

$$Z_0 = \frac{h_{obj}}{\frac{d_{yb}}{d_{zb}} - \frac{d_{yt}}{d_{zt}}}. \quad (18)$$

Аналогичным образом можно получить эту формулу при известной ширине объекта w_{obj} . Разница будет заключаться только в используемых точках: вместо верхней и нижней точки для поиска координат будут использоваться координаты по оси X_c справа и слева: X_r и X_l , как показано на рис. 4.

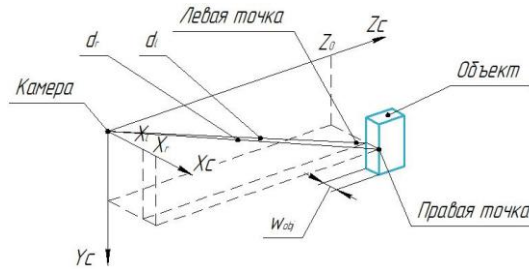


Рис. 4. Вычисление расстояния по известным правой и левой точкам

Итоговая формула вычисления расстояния по известной ширине и координатам точки будет иметь вид (19).

$$Z_0 = \frac{w_{obj}}{\frac{d_{xr}}{d_{zr}} - \frac{d_{xl}}{d_{zl}}}. \quad (19)$$

Описание экспериментального макета. Для экспериментального исследования предлагаемых алгоритмов методом 3D-печати был изготовлен макет системы кругового обзора для установки и последующего использования на мобильном роботе. Чертеж макета и фотография робота с установленным макетом представлены на рис. 5. Высота камер над поверхностью пола составляет 24 см. Положения камер относительно центральной точки макета: левая и правая смещены на 11 см, передняя выдвинута на 17 см, задняя на 8 см.

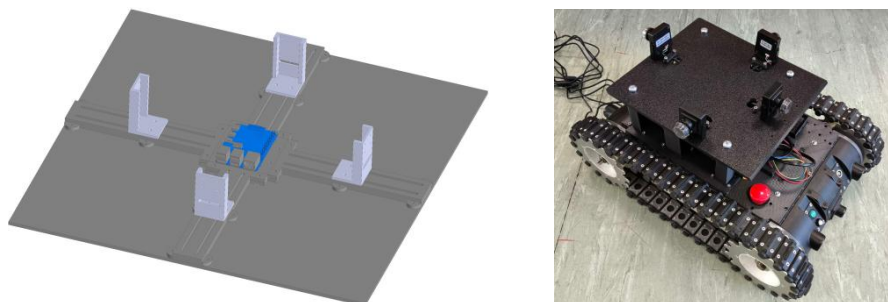


Рис. 5. Чертеж макета системы кругового обзора и фото робота с макетом

Экспериментальное исследование определения положения объектов. Для проверки полученных формул был проведён ряд экспериментов на 3 различных по ширине и высоте объектах. Для этого были сделаны кадры с известным расстоянием до объектов и его размерами, по которым математически было вычислено аналогичное расстояние программными средствами.

Размеры объектов для испытаний представлены в табл. 2.

Таблица 2

Размеры объектов для экспериментального исследования

№ объекта	Высота h_{obj} , см	Ширина w_{obj} , см
1	26,0	17,0
2	40,5	21,5
3	64,5	20,5

Для каждого объекта в каждом положении было сделано по 5 снимков с 4 широкоугольных камер одинаковой модели. Высота камер над полом 24,0 см. Для каждого снимка точки интереса (верхняя, нижняя, правая, левая, центральная) определялись рамкой, в которую помещался предмет (рис. 6).

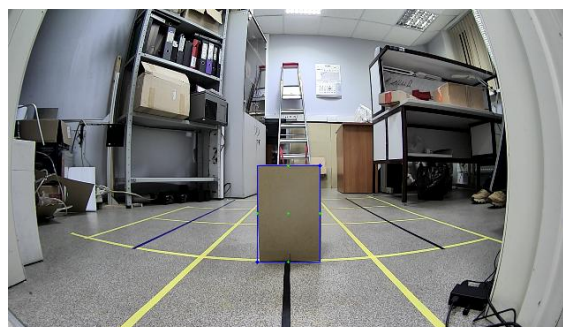


Рис. 6. Пример выделения объекта рамкой с точками интереса

Для оценки точности вычисления положения объекта в поле зрения камеры различными способами был выполнен ряд измерений. Дистанция от камеры до объекта варьировалась от 75 до 225 см, отклонение от оптической оси менялось от 0 до 1,5 м. В первом

эксперименте оценивалось качество определения дистанции до объекта (т.е. Z_0 , расстояние до плоскости, перпендикулярной оптической оси камеры, в которой расположена центральная нижняя точка выделенной рамки). Использовались уравнения, основанные на известной высоте камеры («... по H_c ») и известной ширине и высоте объекта («... по H_o » и «... по W_o »). Результаты экспериментов представлены в табл. 3.

Во втором эксперименте оценивалось качество определения размеров объекта с использованием построенных на основе формул и принципов калибровки геометрических соотношений. Оценивались ширина и высота объекта (« W_o ...» и « H_o ...») также по известной высоте камеры над опорной поверхностью, по заданной ширине объекта и заданной высоте объекта (дистанция и высота по ширине, дистанция и ширина по высоте). Результаты представлены в табл. 4.

Остановимся на метриках, представленных в таблицах. В левой части таблицы в столбцах расположены средние значения величин, полученные по всему набору возможных положений для заданных расстояния или габаритов объектов. Можно рассматривать как математическое ожидание измерения расстояния или размера в поле зрения камеры. В правой части представлено среднеквадратичное (стандартное) отклонение соответствующего измерения. Можно рассматривать его как меру разброса измерений относительно среднего, представленного в левой части.

Таблица 3

Результаты оценки дистанции до объекта разными способами (см.)

Дист.	Z_0 по H_c	Z_0 по W_o	Z_0 по H_o	$\sigma(Z_0/H_c)$	$\sigma(Z_0/W_o)$	$\sigma(Z_0/H_o)$
75	74,31	95,7	76,61	7,9	8,83	7,62
125	117,95	137,17	123,41	5,88	9,46	6,25
175	157,9	176,73	167,39	4,24	8,85	3,81
225	205,89	221,447	214,15	7,5	9,87	5,9

Таблица 4

Результаты оценки размеров объекта разными способами (см.)

Разм. объекта, см	W=17, H=26	W=22, H=40	W=21, H=65
H_o по H_c	25,11	39,06	60,87
W_o по H_c	14,89	18,3	18,07
H_o по W_o	29,17	46,58	69,53
W_o по H_o	15,44	19	19,17
$(H_o \text{ по } H_c)$	0,64	1,05	1,69
$(W_o \text{ по } H_c)$	1,77	2,02	1,49
$(H_o \text{ по } W_o)$	4,48	6,65	6,53
$(W_o \text{ по } H_o)$	1,89	2,22	1,63

Из полученных результатов можно сделать следующие выводы. Наилучшим образом при прочих равных себя показало вычисление дистанции до нижней точки объекта по известной высоте камеры над поверхностью перемещения. Тем не менее, оценка по большему измерению объекта позволяет достигать схожих результатов. Следует отметить, что ошибки определения расстояния до объекта возникают во многом в связи с тем, что детектор стремится выделить объект целиком, захватывая некоторое количество фона, и это вносит искажения в определение положения середины нижней части объекта и вообще его габаритов на изображении.

Что касается оценки метрических размеров объекта, наилучшие результаты также были получены по известной высоте камеры и известной высоте объекта. Это отчасти связано с тем, что для оценки использовались объекты, вертикальное измерение которых существенно больше горизонтального. Отсюда, цена ошибки выделения в 1 пиксель для пересчета в градусы угла при измерении ширины оказывается существенно выше, вырастают и отклонения измерений, и разброс.

С точки зрения обеспечения безопасности движения даже такие ошибки могут быть признаны допустимыми, с учетом того, что они всегда в большую сторону, а быстродействие детектора составляет до 3 измерений в секунду и штатной скорости передвижения около 0,5 м/с (50 см/с).

Пример работы системы безопасности. Демонстрация возможного результата работы предлагаемой системы безопасности приведена на рис. 7. Слева представлены результаты обнаружения объектов, справа маркеры, обозначающие рассчитанные положения объектов на плоскости.

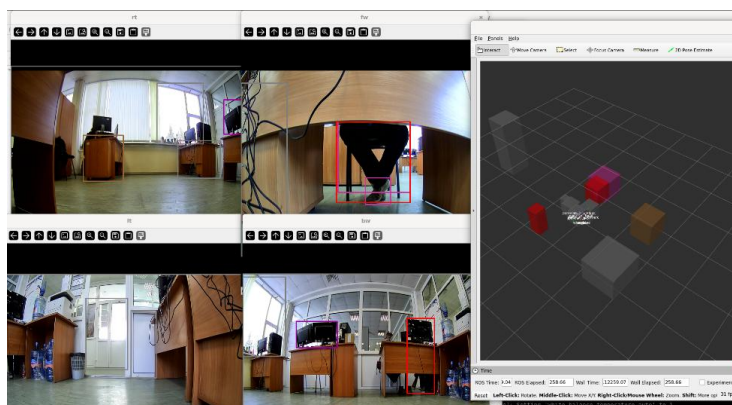


Рис. 7. Пример результатов работы системы безопасности

С учетом того, что используемая нейронная сеть обучена на классах набора данных OpenImages v7, и она способна распознавать 600 различных категорий объектов, для простоты отображения категории были объединены по цветам. В частности, из видимого на примере, красным обозначены люди (возможные подвижные объекты), сиреневым – обувь и одежда (часто бывают связаны с людьми), светло-коричневым – мебель, серым – различные объекты, которые могут быть расценены как препятствия.

Заключение. В работе представлена реализация системы обеспечения безопасности перемещения для мобильного робототехнического или иного транспортного средства, оснащенного системой кругового обзора на основе нескольких широкоугольных камер с суммарным полем зрения всех камер по горизонтали в 360 градусов и более. Так как в мобильной робототехнике есть существенные ограничения на энергопотребление, габариты и массу вычислителя, был выбран микрокомпьютер Orange Pi 5 Max с TPU на базе Rockchip RK3588. Его достаточно для предобработки изображений, расчета координат объектов и визуализации. Тензорные ядра обеспечивают успешный расчет нейронных сетей.

Были рассмотрены несколько вариантов предобработки изображений и вычисления нейронной сети. По результатам экспериментов выбран вариант с последовательным вызовом обзора каждого из изображений, т.к. сжатие показало существенное падение числа обнаруживаемых объектов, а прочие варианты полнокадровой обработки заняли больше времени. Получение данных о координатах объектов осуществляется с частотой в среднем 1,05 с.

Рассмотрен принцип построения калибровки Скарамузза для камер «рыбий глаз», после чего на основе известной калибровки и несложных геометрических соотношений предложено два варианта алгоритма оценки параметров объекта. В первом случае неизвестные параметры вычисляются по известной пиксельной высоте центра основания ограничивающей рамки и метрической высоте камеры над поверхностью, во втором случае – по пиксельным размерам рамки и геометрическим размерам объекта. По данным экспериментального исследования предложенных алгоритмов наибольшую точность по определению расстояния и размеров показали расчеты на основе высоты камеры над поверхностью (стандартное отклонение от 4,2 до 7,9 мм) и на основе высоты объекта (отклонение от 3,8 до 7,6 мм). Наилучшую точность в оценке расстояния также показали

алгоритмы на основе известной высоты камеры над поверхностью (стандартное отклонение от 0,64 до 2,02 мм) и по известной высоте объекта (от 1,63 до 2,22 мм). Оценка по ширине также может быть использована для обеспечения безопасности, если иная информация отсутствует.

Предложенный подход к обеспечению безопасности передвижения автономного транспортного средства на основе детектора объектов и камер системы кругового обзора признан работоспособным. В качестве дальнейшего развития работы предполагается провести исследование различных алгоритмов калибровки, внести в расчет информацию об ориентации транспортного средства по данным гироскопа, рассмотреть другие архитектуры нейронных сетей и обучить набор весов, адаптированный под решаемые задачи.

Результаты получены в рамках выполнения государственного задания Минобрнауки России № 075-00558-26-00 от 12.01.2026 «Исследование методов анализа слабоструктурированных данных, обработки знаний и создания когнитивных агентов на базе комбинированных глубоких нейронных сетей» (FNRG-2025-0008 1024050200009-5-1.2.1;2.2.2).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Половко С.А., Смирнова Е.Ю., Степанов Д.Н. Интеллектуальные системы технического зрения для безопасности и навигации // Мехатроника, автоматизация, управление. – 2009. – №. 3. – С. 33-39.
2. Asvadi A., et al. 3D Lidar-based static and moving obstacle detection in driving environments: An approach based on voxels and multi-region ground planes // Robotics and Autonomous Systems. – 2016. – Vol. 83. – P. 299-311.
3. Sekine T. The image of automobile safety // International Association of Traffic and Safety Sciences, Interdisciplinary Wisdom of IATSS. – 2015. – P. 62.
4. Hamsini S., Kathires M. Automotive safety systems // Automotive Embedded Systems: Key Technologies, Innovations, and Applications. – Cham: Springer International Publishing, 2021. – P. 1-18.
5. Kazak A.N., et al. Service robotics development trends in the Russian Federation // AIP Conference Proceedings. – AIP Publishing LLC, 2021. – Vol. 2402, No. 1. – P. 070020.
6. Scaramuzza D., Martinelli A., Siegwart R. A toolbox for easily calibrating omni-directional cameras // 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems. – IEEE, 2006. – P. 5695-5701.
7. Park K.T., et al. Training YOLOv8 Object Detection Model with Synthetic Images // 2024 24th International Conference on Control, Automation and Systems (IC-CAS). – IEEE, 2024. – P. 524-528.
8. OpenImages V7 Description. – URL: https://storage.googleapis.com/openimages/web/factsfigures_v7.html (дата обращения: 06.02.2026).
9. Meng M., et al. Distortion-aware room layout estimation from a single fisheye image // 2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). – IEEE, 2021. – P. 441-449.
10. Meng M., et al. Distortion-aware room layout estimation from a single fisheye image // 2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). – IEEE, 2021. – P. 441-449.
11. Yan F., et al. Monocular catadioptric panoramic depth estimation via improved end-to-end neural network model // Frontiers in Neurorobotics. – 2023. – Vol. 17. – P. 1278986.
12. Simioni E., et al. Geometrical calibration for the panover: A stereo omnidirectional system for planetary rover // The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. – 2020. – Vol. 43. – P. 1151-1158.
13. Coates A.J., et al. The PanCam instrument for the ExoMars rover // Astrobiology. – 2017. – Vol. 17, No. 6-7. – P. 511-541.
14. Ullah H., et al. Automatic 360 mono-stereo panorama generation using a cost-effective multi-camera system // Sensors. – 2020. – Vol. 20, No. 11. – P. 3097.
15. Cui H., Zhao Z., Zhang F. Research on Panorama Generation from a Multi-Camera System by Object-Distance Estimation // Applied Sciences. – 2023. – Vol. 13, No. 22. – P. 12309.
16. Прудников Н.В., Шлишевский В.Б. Панорамные оптико-электронные устройства кругового и секторного обзора // Вестник СГУГиТ. – 2016. – №. 1 (33). – С. 148-161.
17. Kakani V., Kim H. Adaptive self-calibration of fisheye and wide-angle cameras // TENCON 2019-2019 IEEE Region 10 Conference (TENCON). – IEEE, 2019. – P. 976-981.
18. Douskos V., Kalisperakis I., Karras G. Automatic calibration of digital cameras using planar chessboard patterns // Proceedings of the 8th Conference on Optical. – 2007. – P. 9-12.
19. Jiang G., Quan L. Detection of concentric circles for camera calibration // Tenth IEEE International Conference on Computer Vision (ICCV'05). Vol. 1. – IEEE, 2005. – P. 333-340.

20. Kannala J., Brandt S.S. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses // *IEEE transactions on pattern analysis and machine intelligence*. – 2006. – Vol. 28, No. 8. – P. 1335-1340.
21. Liu W., et al. Ssd: Single shot multibox detector // *European conference on computer vision*. – Cham: Springer International Publishing, 2016. – P. 21-37.
22. Duan K., et al. Centernet: Keypoint triplets for object detection // *Proceedings of the IEEE/CVF international conference on computer vision*. – 2019. – P. 6569-6578.
23. Alhasanat M.N., et al. Retinanet-based approach for object detection and distance estimation in an image // *International Journal on Communications Antenna and Propagation (IRECAP)*. – 2021. – Vol. 11, No. 1. – P. 1-9.
24. Tan M., Pang R., Le Q.V. Efficientdet: Scalable and efficient object detection // *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. – 2020. – P. 10781-10790.
25. Redmon J., Divvala S., Girshick R., Farhadi A. You Only Look Once: Unified, Real-Time Object Detection // *arXiv:1506.02640 [cs.CV]*. – 2015. – URL: <https://arxiv.org/abs/1506.02640> (дата обращения: 13.02.2026).
26. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks // *arXiv:1506.01497 [cs.CV]*. 2015. – URL: <https://arxiv.org/pdf/1506.01497> (дата обращения: 06.02.2026).
27. He K., Gkioxari G., Dollár P., Girshick R. Mask R-CNN // *arXiv:1703.06870 [cs.CV]*. – 2017. – URL: <https://arxiv.org/pdf/1703.06870> (дата обращения: 07.02.2026).
28. Sun P., et al. Sparse r-cnn: End-to-end object detection with learnable proposals // *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. – 2021. – P. 14454-14463.
29. Shridhar A., Tomson P., Innes M. Interoperating deep learning models with onnx. jl // *Proceedings of the JuliaCon Conferences*. – 2020. – Vol. 1, No. 1. – P. 59.
30. Jegham N., et al. Yolo evolution: A comprehensive benchmark and architectural review of yolov12, yolov11, and their previous versions // *arXiv preprint arXiv:2411.00201*. – 2024.
31. Ultralytics | Revolutionizing the World of Vision AI. – URL: <https://www.ultralytics.com/> (дата обращения 02.02.2026).
32. GitHub – Pelochus/ezrknn-toolkit2. Making Rockchip's RKNN-Toolkit-2 easier for SBCs like Orange Pi 5 or Radxa Rock 5. – URL: <https://github.com/Pelochus/ezrknn-toolkit2> (дата обращения 02.02.2026).

REFERENCES

1. Polovko S.A., Smirnova E.Yu., Stepanov D.N. Intellektual'nye sistemy tekhnicheskogo zreniya dlya bezopasnosti i navigatsii [Intelligent machine vision systems for safety and navigation], *Mekhatronika, avtomatizatsiya, upravlenie* [Mechatronics, Automation, Control], 2009, No. 3, pp. 33-39.
2. Asvadi A., et al. 3D Lidar-based static and moving obstacle detection in driving environments: An approach based on voxels and multi-region ground planes, *Robotics and Autonomous Systems*, 2016, Vol. 83, pp. 299-311.
3. Sekine T. The image of automobile safety, *International Association of Traffic and Safety Sciences, Interdisciplinary Wisdom of IATSS*, 2015, pp. 62.
4. Hamsini S., Kathiresh M. Automotive safety systems, *Automotive Embedded Systems: Key Technologies, Innovations, and Applications*. Cham: Springer International Publishing, 2021, pp. 1-18.
5. Kazak A.N., et al. Service robotics development trends in the Russian Federation, *AIP Conference Proceedings*. AIP Publishing LLC, 2021, Vol. 2402, No. 1, pp. 070020.
6. Scaramuzza D., Martinelli A., Siegwart R. A toolbox for easily calibrating omni-directional cameras, *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 5695-5701.
7. Park K.T., et al. Training YOLOv8 Object Detection Model with Synthetic Images, *2024 24th International Conference on Control, Automation and Systems (IC-CAS)*. IEEE, 2024, pp. 524-528.
8. OpenImages V7 Description. Available at: https://storage.googleapis.com/openimages/web/factsfigures_v7.html (accessed 6 February 2011).
9. Meng M., et al. Distortion-aware room layout estimation from a single fisheye image, *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2021, pp. 441-449.
10. Meng M., et al. Distortion-aware room layout estimation from a single fisheye image, *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2021, pp. 441-449.
11. Yan F., et al. Monocular catadioptric panoramic depth estimation via improved end-to-end neural network model, *Frontiers in Neurorobotics*, 2023, Vol. 17, pp. 1278986.
12. Simioni E., et al. Geometrical calibration for the panover: A stereo omnidirectional system for planetary rover, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2020, Vol. 43, pp. 1151-1158.

13. Coates A.J., et al. The PanCam instrument for the ExoMars rover, *Astrobiology*, 2017, Vol. 17, No. 6-7, pp. 511-541.
14. Ullah H., et al. Automatic 360 mono-stereo panorama generation using a cost-effective multi-camera system, *Sensors*, 2020, Vol. 20, No. 11, pp. 3097.
15. Cui H., Zhao Z., Zhang F. Research on Panorama Generation from a Multi-Camera System by Object-Distance Estimation, *Applied Sciences*, 2023, Vol. 13, No. 22, pp. 12309.
16. Prudnikov N.V., Shlishevskiy V.B. Panorамные оптико-электронные устройства кругового и секторного обзора [Panoramic optoelectronic devices for circular and sectorial surveillance], *Vestnik SGUGiT* [Bulletin of SSUGT], 2016, No. 1 (33), pp. 148-161.
17. Kakani V., Kim H. Adaptive self-calibration of fisheye and wide-angle cameras, *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*. IEEE, 2019, pp. 976-981.
18. Douskos V., Kalisperakis I., Karras G. Automatic calibration of digital cameras using planar chess-board patterns, *Proceedings of the 8th Conference on Op-tical*, 2007, pp. 9-12.
19. Jiang G., Quan L. Detection of concentric circles for camera calibration, *Tenth IEEE International Conference on Computer Vision (ICCV'05)*. Vol. 1. IEEE, 2005, pp. 333-340.
20. Kannala J., Brandt S.S. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses, *IEEE transactions on pattern analysis and machine intelligence*, 2006, Vol. 28, No. 8, pp. 1335-1340.
21. Liu W., et al. Ssd: Single shot multibox detector, *European conference on computer vision*. Cham: Springer International Publishing, 2016, pp. 21-37.
22. Duan K., et al. Centernet: Keypoint triplets for object detection, *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6569-6578.
23. Alhasanat M.N., et al. Retinanet-based approach for object detection and dis-tance estimation in an image, *International Journal on Communications Antenna and Propagation (IRECAP)*, 2021, Vol. 11, No. 1, pp. 1-9.
24. Tan M., Pang R., Le Q.V. Efficientdet: Scalable and efficient object de-tection, *Proceedings of the IEEE/CVF conference on computer vision and pat-tern recognition*, 2020, pp. 10781-10790.
25. Redmon J., Divvala S., Girshick R., Farhadi A. You Only Look Once: Unified, Real-Time Ob-ject Detection, *arXiv:1506.02640 [cs.CV]*, 2015. Available at: <https://arxiv.org/abs/1506.02640> (accessed 13 February).
26. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, *arXiv:1506.01497 [cs.CV]*, 2015. Available at: <https://arxiv.org/pdf/1506.01497> (accessed 06 February 2026).
27. He K., Gkioxari G., Dollár P., Girshick R. Mask R-CNN // *arXiv:1703.06870 [cs.CV]*, 2017. Availa-ble at: <https://arxiv.org/pdf/1703.06870> (accessed 7 February).
28. Sun P., et al. Sparse r-cnn: End-to-end object detection with learnable proposals, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 14454-14463.
29. Shridhar A., Tomson P., Innes M. Interoperating deep learning models with onnx. JI, *Proceedings of the JuliaCon Conferences*, 2020, Vol. 1, No. 1, pp. 59.
30. Jegham N., et al. Yolo evolution: A comprehensive benchmark and archi-tectural review of yolov12, yolo11, and their previous versions, *arXiv preprint arXiv:2411.00201*, 2024.
31. Ultralytics | Revolutionizing the World of Vision AI. Available at: <https://www.ultralytics.com/> (дата обращения 02.02.2026).
32. GitHub – Pelochus/ezrknn-toolkit2. Making Rockchip’s RKNN-Toolkit-2 easier for SBCs like Orange Pi 5 or Radxa Rock 5. Available at: <https://github.com/Pelochus/ezrknn-toolkit2> (accessed 2 February).

Фомин Иван Сергеевич – Государственный научный центр РФ – Федеральное государственное автономное научное учреждение «Центральный научно-исследовательский и опытно-конструкторский институт робототехники и технической кибернетики»; e-mail: i.fomin@rtc.ru; г. Санкт-Петербург, Россия; тел.: +78125523351; научный сотрудник.

Базельцев Артемий Андреевич – Санкт-Петербургский политехнический университет Петра Великого»; e-mail: bazeltsev.aa@edu.spbstu.ru; г. Санкт-Петербург, Россия; тел.: +78007071899; студент.

Fomin Ivan Sergeevich – Russian State Scientific Center for Robotics and Technical Cybernetics; e-mail: i.fomin@rtc.ru; Saint Petersburg, Russia; phone: +78125523351; researcher.

Baseltsev Artemii Andreevich – Peter the Great St.Petersburg Polytechnic University; e-mail: bazeltsev.aa@edu.spbstu.ru; Saint-Petersburg, Russia; phone: +78007071899; undergraduate student.