

УДК 004:658.5

DOI 10.17513/snt.40610

МЕТОДИКА ЛОКАЛИЗАЦИИ ПОСАДОЧНЫХ ЯЧЕЕК ДЛЯ МОБИЛЬНЫХ РОБОТОВ-МАНИПУЛЯТОРОВ

Кушнир А.П., Макаров М.А., Хоменко С.Н.

*Федеральное государственное бюджетное образовательное учреждение высшего образования
«МИРЭА – Российский технологический университет», Москва, Российская Федерация,
e-mail: MakMak95@yandex.ru*

В статье предложена методика трехмерного машинного зрения для роботизированных участков многономенклатурного мелкосерийного производства. Цель исследования – повышение адаптивности, гибкости и мобильности производства на основе разработки метода машинного зрения, позволяющего оперативно передавать роботу-манипулятору точные координаты заготовок и инструмента при смене номенклатуры выпускаемых изделий. Научная новизна подхода заключается в первичном обнаружении не самих объектов, а их посадочных ячеек, обозначенных стабильными фиксирующими ограничителями с высококонтрастными AprilTag-метками. Эти метрические ориентиры образуют опорную сетку, относительно которой выполняются все дальнейшие измерения, что исключает зависимость от формы и оптических свойств деталей, уменьшает число ложных срабатываний и существенно упрощает переналадку. Методика использует адаптивное фоновое вычитание на основе модели смеси гауссовых распределений с локальной контурной или нейросетевой детекцией объектов и стереозрением с динамической калибровкой типа eye-in-hand, реализованной средствами Robot Operating System. Экспериментальные испытания подтвердили устойчивость алгоритма при недостаточных условиях освещенности и частых сменах номенклатуры изделий, а также высокую точность локализации и производительность обработки данных. Полученные результаты демонстрируют практическую эффективность предложенной методики и возможность ее применения для гибкой автоматизации манипуляционных задач в условиях многономенклатурного мелкосерийного производства.

Ключевые слова: машинное зрение, стереокамера, AprilTag, посадочная ячейка, hand-eye, калибровка, робот-манипулятор

METHOD OF LOCALIZING LANDING CELLS FOR MOBILE ROBOTIC MANIPULATORS

Kushnir A.P., Makarov M.A., Khomenko S.N.

*Federal State Budgetary Educational Institution of Higher Education
“MIREA – Russian Technological University”, Moscow, Russian Federation,
e-mail: MakMak95@yandex.ru*

The article proposes a three-dimensional machine vision technique for robotic sections of multi-product small-batch production. The aim of the study is to increase the adaptability, flexibility, and mobility of production based on the development of a machine vision method that allows the robot manipulator to quickly receive the exact coordinates of workpieces and tools when changing the range of products manufactured. The scientific novelty of the approach lies in the initial detection not of the objects themselves, but of their mounting cells, marked with stable fixing restraints with high-contrast AprilTag marks. These metric landmarks form a reference grid relative to which all further measurements are performed, which eliminates dependence on the shape and optical properties of parts, reduces the number of false positives, and significantly simplifies retooling. The method uses adaptive background subtraction based on a Gaussian mixture model with local contour or neural network object detection and stereo vision with dynamic eye-in-hand calibration implemented by the Robot Operating System. Experimental tests confirmed the stability of the algorithm under insufficient lighting conditions and frequent changes in the product range, as well as high localization accuracy and data processing performance. The results demonstrate the practical effectiveness of the proposed methodology and its applicability for flexible automation of handling tasks in multi-product small-batch production.

Keywords: computer vision, stereo camera, AprilTag, landing cell, hand-eye, calibration, robot manipulator

Введение

Современное мелкосерийное производство, характеризующееся частой сменой номенклатуры изделий, требует от роботизированных технологических комплексов (РТК) высокой степени адаптивности, гибкости и мобильности. Ключевой задачей становится обеспечение точной локализации координат заготовок и технологической оснастки на рабочих столах с последующей привязкой к системе координат робота-ма-

нипулятора. Особенную важность данная задача приобретает для мобильных комплексов, положение которых не является фиксированным в пространстве и носит случайный характер. Под «мобильным комплексом» далее понимается конструкция из мобильной платформы с установленным на ней роботом-манипулятором. В этом случае необходимо постоянно осуществлять и корректировать привязку систем координат мобильного робота-манипуля-

тора не только с координатами заготовок, но и с расположением самих рабочих столов с манипулируемыми технологическими объектами. Применение предложенной методики автоматической калибровки и детекции позволяет минимизировать ручное вмешательство, существенно снизить время переналадки, повысить гибкость и мобильность технологического процесса.

Анализ существующих результатов исследований показывает, что значительное внимание авторы уделяют методам повышения точности позиционирования и калибровки роботизированных систем. В работе V. Burde и соавт. [1] предлагается, например, использовать автоматическую калибровку рабочего пространства на основе гомографического матричного преобразования. Недостатком данного метода является отсутствие поддержки обнаружения в условиях недостаточного освещения. H.-N. Nguyen и его коллеги представили гибридный метод калибровки, объединяющий идентификацию модели робота и компенсацию ошибок на основе использования нейронных сетей [2], но без учета изменчивости производственных условий. Схожие задачи рассмотрены в работах D. Allegro и Z. Jiang, в которых предлагается метод многокамерной калибровки с камерой, установленной на манипуляторе («eye-in-hand») [3, 4]. Следует отметить, что данный метод не предусматривает быстрой переналадки при смене номенклатуры выпускаемых изделий, поскольку жестко зависит от конкретной конфигурации камер на манипуляторе и не устраняет необходимость присутствия человека в ходе данной операции. Также стоит упомянуть подробный научный обзор методов калибровки для роботизированных систем I. Enebusе и коллектива, включающий классические, оптимизационные и современные методы вариации [5]. Однако стоит отметить, что данный обзор, несмотря на обширный охват различных методов калибровки, основан преимущественно на детерминированном подходе. При этом в обзоре недостаточно подробно рассмотрена проблема асинхронности передачи данных при онлайн-калибровке, а также влияние помех и шумов, что снижает применимость результатов для критичных к точности систем.

Важным направлением исследований является разработка алгоритмов машинного обучения и компьютерного зрения. J. Wang и его команда рассмотрели особенности визуального восприятия и оценки глубины при роботизированной сварке [6], однако метод чувствителен к изменению освещенности и поэтому не может быть

использован для мобильных роботов-манипуляторов, которым приходится работать в переменных условиях. M. Li с коллегами представили систему управления роботизированной сваркой с улучшенным алгоритмом YOLOv5 и датчиками глубины [7]. Существенным ограничением данного метода являются стационарные условия работы установки, так как в предложенном решении не учтены вопросы переналадки системы при изменении производственной задачи. В исследовании G. Velickaite также отмечено, что замена детекторов меток на глубокие модели семейства YOLO повышает устойчивость распознавания меток при засвете и частичных окклюзиях, что облегчает идентификацию по классам [8]. Недостатком данного решения является зависимость от размеченных данных и условий съемки, а также необходимость переобучения при изменении сцены и набора меток.

Различные варианты калибровки подробно рассмотрены в работе коллектива Y. Zhou [9], где для задач захвата и перемещения изделий реализован способ внешней калибровки без использования меток, основанный на серии демонстрационных захватов с одновременным восстановлением внешних параметров камеры и роботизированной системы. Способ обеспечивает высокую точность применительно к стационарным ячейкам при фиксированной конфигурации камер и оснастки, но предполагает разовую настройку под конкретную производственную задачу и требует повторения процедуры при существенном изменении сцены. В отличие от такого подхода, перспективным представляется комбинированный вариант, в котором система AprilTag-меток используется как устойчивый «якорь» для ускоренной переналадки системы машинного зрения в роботизированном технологическом комплексе. Рабочие позиции, представленные посадочными ячейками, кодируются кластерами меток, а уточняющая детекция объектов внутри ячеек выполняется методами без использования меток, включая контурную и нейросетевую детекции. За счет этого калибровка «камера-манипулятор» и положение стола в базовой системе координат робота остаются неизменными, а смена номенклатуры выпускаемых изделий сводится к механической перестановке ячеек с метками без повторной калибровки сцены.

В работе M.B. Егорцева описывается практическая и доступная система машинного зрения с применением одной камеры и использованием алгоритма SURF (Speeded-Up Robust Features) [10]. Недостатком предложенной системы является зависимость

от одиночной камеры без глубины, чувствительной к окклюзиям. Кроме того, распознавание объектов с применением алгоритма SURF затруднительно по отношению к слаботекстурным и блестящим объектам. Проблематика отражающих, слаботекстурных поверхностей и устойчивости к дефектам освещения также подробно систематизирована в научном обзоре А. Ibrahim в рамках промышленного визуального контроля [11].

Различные методы локализации и захвата объектов исследовались К. Kimble и коллегами, которые предложили стандарты для оценки роботизированных систем сборки мелких деталей [12], однако вопросы оперативной переналадки в данной работе остались без внимания. В исследовании С.-Н. Tsai с соавт. предложен метаметод позиционирования, основанный на машинном обучении [13]. Недостаток метода состоит в том, что результат обучения может быть применим только для рассмотренного класса заготовок и не учитывает необходимость оперативной адаптации к новым типам объектов. При каждом изменении номенклатуры обрабатываемых деталей необходимо проводить повторные исследования, что снижает производительность технологического процесса.

В прикладных задачах комплектования коллективы G. Boschetti и M. Ojer представили фреймворки и приемы для взаимодействия человека и робота, демонстрирующие прирост производительности. Вместе с тем в работах [14, 15] не отражены вопросы надежной локализации и быстрой переналадки производства.

Z. Wang и A. Westman использовали технологию SLAM (simultaneous localization and mapping) – одновременную локализацию и построение карты на основе меток, устойчивую к шумам и искажениям [16, 17], но не затронули задачу переналадки.

В работе А. Chen и его группы описан адаптивный метод оптимизации захвата объектов с использованием датчиков глубины [18], однако предложенное решение не предусматривает автоматическое переобучение при смене заготовок. N. Guo с коллегами разработали метод оценки позы объекта и адаптации захвата на основе анализа геометрии и облака точек, но оставили без внимания автоматическую переналадку технологической системы [19].

Таким образом, проведенный анализ показал, что большинство существующих подходов направлено на решение узкоспециализированных задач обнаружения, калибровки и захвата объектов определенной формы без учета комплексной проблемы оперативной переналадки системы машинного зрения

при смене номенклатуры изделий, переменных координат самого робота-манипулятора и недостаточных условиях освещенности рабочей зоны. Это затрудняет интеграцию таких решений в мелкосерийное многономенклатурное производство и приводит к снижению производительности всей системы в целом. На этом фоне актуальны решения, объединяющие устойчивую локализацию с использованием меток, глубинную оценку позы и трансформацию координат в базовую систему робота с целью повышения надежности в реальных промышленных условиях и уменьшения зависимости от фиксированных конфигураций.

Таким образом, традиционная методика обнаружения и идентификации объектов на основе меток включает в себя несколько уровней. На первом уровне кластеры AprilTag-меток задают устойчивую метрическую сетку рабочих позиций, на втором уровне внутри каждой ячейки выполняется детекция занятости и уточняющее определение позы заготовки с помощью контурного или нейросетевого детекторов, а передача координат в систему управления осуществляется через калибровку «камера на манипуляторе» и дерево трансформаций TF2 в режиме реального времени. При этом подход на основе меток связан с эксплуатационными издержками. Во-первых, метки полностью разделяют задачу координатной привязки и геометрические или оптические свойства заготовок, поэтому независимо от формы деталей система всегда распознает одни и те же опорные кластеры. Во-вторых, переналадка сводится к механическому изменению конфигурации ячеек и оснастки с заранее раскодированными метками, без повторной демонстрации траекторий, переобучения нейросетевых моделей или сложной переоценки внешних параметров камер. В-третьих, стоимость и трудоемкость наклеек и обслуживания AprilTag-меток практически не зависят от числа типов обрабатываемых изделий, тогда как для методов без использования меток расходы на сбор и разметку данных возрастают с ростом номенклатуры.

Для преодоления указанных недостатков и обеспечения оперативной переналадки машинного зрения для мобильных РТК в условиях многономенклатурного мелкосерийного производства авторами предлагается метод, реализующий автоматизированную локализацию объектов и технологической оснастки на основе системы AprilTag-меток с последующим точным обнаружением и пространственной привязкой объектов, минимизирующий объем переналадки при изменении производственной задачи.

С учетом рассмотренных работ научный вклад данной статьи можно сформулировать как метод ускоренной переналадки системы машинного зрения для мобильных РТК на основе иерархической детекции с использованием AprilTag-меток и динамической калибровки через TF2.

Цель исследования – повышение адаптивности, гибкости и мобильности производства на основе разработки метода машинного зрения, позволяющего оперативно передавать роботу-манипулятору точные координаты заготовок и инструмента при смене номенклатуры выпускаемых изделий.

Для достижения поставленной цели были решены следующие задачи:

1. Разработана подсистема обнаружения посадочных ячеек на основе высококонтрастных AprilTag-меток, позволяющих исключить зависимость от геометрии и оптических свойств перемещаемых объектов.

2. Обеспечено стабильное распознавание занятости посадочных ячеек, независимо от условий освещения, светоотражающих свойств поверхности, геометрии и фактуры перемещаемых объектов.

3. Реализован адаптивный подход для уточняющей локальной детекции деталей с использованием классических алгоритмов обработки изображений и нейросетевого подхода на основе архитектуры YOLOv8.

4. Разработана подсистема преобразования координат из системы технического зрения в систему координат робота-манипулятора, не требующей дополнительной калибровки при смене номенклатуры выпускаемых изделий.

5. Повышена надежность обнаружения и идентификации объектов путем реализации механизма подавления повторных детекций целей для предотвращения повторной обработки объектов.

Материалы и методы исследования

Предлагаемый метод компьютерного зрения предназначен для роботизированных рабочих мест мелкосерийного производства, где номенклатура изделий часто меняется. Основное допущение заключается в том, что каждая посадочная позиция изделия обозначена группой высококонтрастных AprilTag-меток, установленных на фиксированных ограничителях рабочих столов. Такая схема позволяет перенести фокус детекции с непредсказуемой геометрии самих заготовок на стабильные метрически калиброванные «якоря» сцены и тем самым снизить ложные срабатывания при смене продукции. Метод реализован как ROS2-узел с пятью взаимосвязанными подсистемами (рис. 1).

Первой срабатывает подсистема ROI-маскирования. На этапе инициализации из YAML-конфигурации загружается бинарная маска областей интереса (ROI), фиксирующая границы рабочей поверхности. Для мобильного робота маска формируется в «канонической» системе стола и при каждом кадре переносится в изображение по текущей гомографии $H_{cam \leftarrow table}$, чтобы положение ROI оставалось корректным при сдвиге мобильной платформы. Маска применяется покадрово к RGB-потoku, благодаря чему компоненты механизма, находящиеся вне зоны захвата, не влияют на последующую обработку изображения [20].

После маскирования к кадру применяется подсистема фоновой вычитки MOG2, поддерживающая сцену с медленно меняющимся освещением и формирующая двухклассовую карту «задний план / передний план». Использовалась адаптивная смесь гауссовых распределений по методу Zivkovic [21], автоматически подбирающая число компонент для каждого пикселя. Конфигурация уровня воспроизводимости включала в себя следующие параметры:

history = 500, определяет количество предыдущих кадров, используемых для обновления фоновой модели;

varThreshold = 25, задает порог чувствительности к изменениям интенсивности пикселей;

detectShadows = true, обеспечивает идентификацию теней с последующим их исключением из области переднего плана.

Значения параметров были выбраны по предварительному набору на «пустой сцене».

Далее включается подсистема обнаружения AprilTag-кластеров. Поток бинаризованных данных накладывается на синхронную глубинную карту стереокамеры. В каждом кадре система ищет априорно известные ID-комбинации AprilTag, образуя устойчивые геометрические кластеры. Для повышения устойчивости к частичным окклюзиям гомография плоскости стола $H_{cam \leftarrow table}$ оценивается методом RANSAC по вершинам детектированных меток [22]. Совокупность обнаруженных меток задает координаты посадочной ячейки $C_i = \{X_i, Y_i, Z_i, w_i, h_i, \theta_i\}$.

После получения координат посадочной ячейки запускается подсистема проверки заполненности ячейки. Для каждой ячейки C_i используется локальный прямоугольный ROI, параметризованный центром, размерами и ориентацией. Ячейка помечается «занятой», если медианная глубина точек внутри ROI превышает адаптивный порог z_{th} , либо площадь пикселей переднего плана в ROI превосходит порог a_{th} .

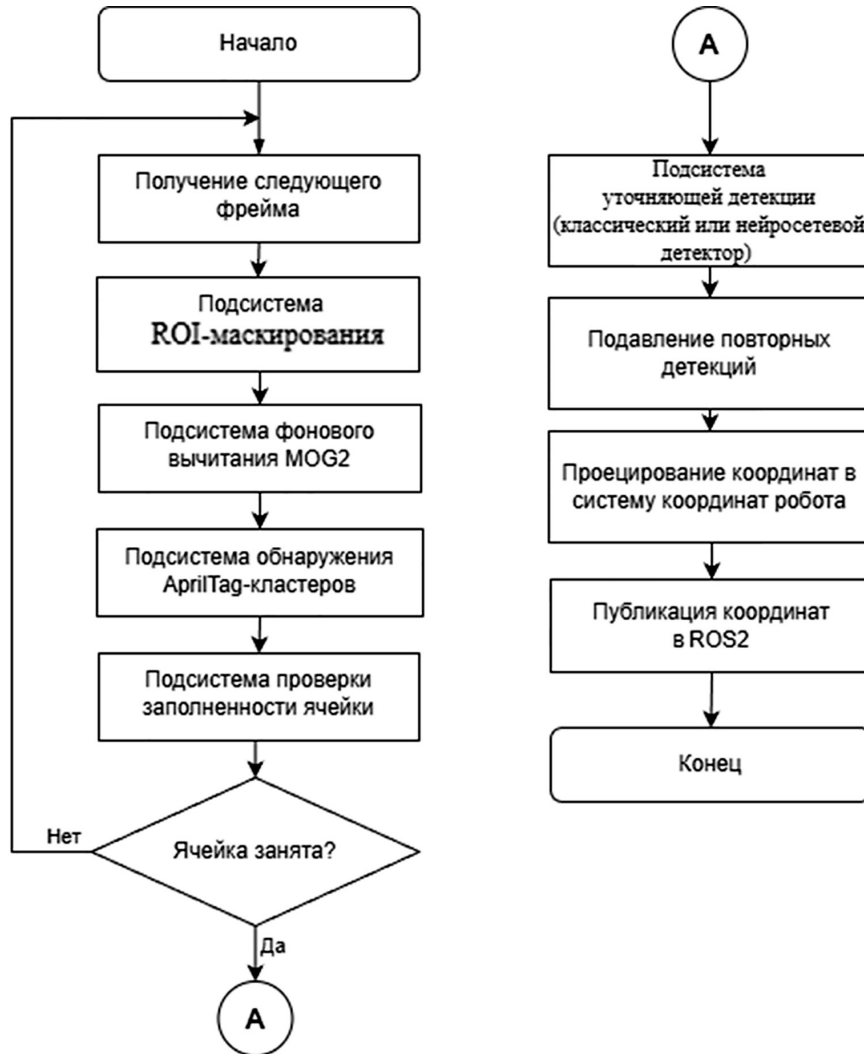


Рис. 1. Блок-схема алгоритма метода компьютерного зрения
Примечание: составлен авторами по результатам данного исследования

Пороговые значения подбираются адаптивно из «эталоны пустой ячейки»: z_{th} – по экспоненциально-взвешенной статистике глубины с запасом, a_{th} – по верхнему перцентилю площади переднего плана. Блики и аномалии подавляются морфологией и временным гистерезисом по N кадрам. Пусть z_t – медианная глубина внутри ROI на t-м кадре для заведомо пустой ячейки. Тогда экспоненциально-взвешенное среднее $\mu_z(t)$ и дисперсия по глубине $\sigma_z^2(t)$ вычисляются по рекуррентным соотношениям:

$$\mu_z(t) = (1 - \alpha) \mu_z(t-1) + \alpha z_t,$$

$$\sigma_z^2(t) = (1 - \alpha) \sigma_z^2(t-1) + \alpha (z_t - \mu_z(t))^2,$$

где $0 < \alpha < 1$ – коэффициент сглаживания (в работе $\alpha = 0,05$).

Порог по высоте задается как

$$z_{th} = \mu_z^{(t)} + k \sigma_z^{(t)}.$$

При значении $k = 2$ (в предположении нормальной формы распределения) ожидаемая доля ложных срабатываний не выше 1–2% на пустой ячейке. Аналогично порог по площади a_{th} задается как эмпирический p-перцентиль распределения площади переднего плана для пустых ячеек, вычисляемый по скользящему окну длиной M кадров. Переоценка порогов выполняется в фоновом режиме раз в N кадров при условии, что ячейка в течение окна наблюдений не содержит детектированных объектов.

Последней в алгоритме является подсистема уточняющей детекции и проекции координат. Если ROI ячейки признана «занятой», внутри нее локально срабатывает либо

контурный детектор, включающий в себя адаптивную пороговую обработку, поиск контуров и аппроксимацию минимальным вписанным прямоугольником, либо нейросетевой детектор на основе YOLOv8-OBV. Центры рамок (u, v, Z_{com}) при этом переводятся в систему координат камеры:

$$P_{cam} = Z_{cam} K^{-1} [u, v, 1]^T,$$

затем – в базовую систему координат робота:

$$P_{base} = T_{base \leftarrow cam} [P_{cam}, 1]^T.$$

Инициализация, включающая в себя внутреннюю калибровку камеры на манипуляторе [13], выполняется один раз, далее актуальное преобразование $T_{base \leftarrow cam}$ публикуется TF2 в режиме реального времени. В подсистеме блока подавления повторных детекций полученный вектор P_t сравнивается с последними n опубликованными центрами, и, если расстояние $\|P_t - P_{t-1}\|$ меньше эмпирического порога, цель считается уже обслуженной и повторно не публикуется.

Результаты исследования и их обсуждение

Предлагаемый метод реализован в виде модуля программного обеспечения, интегрированного в среду Robot Operating System версии ROS2 Humble.

Для обеспечения требуемой точности пространственной локализации заготовок и технологической оснастки необходимо исключить из анализа кадра области, не относящиеся к рабочей зоне РТК. Эта задача решается с помощью подсистемы маскирования областей интереса. На этапе запуска программного модуля проводится загрузка заранее подготовленной бинарной маски из конфигурационного файла формата YAML. Полученную маску применяют к каждому новому кадру, получаемому от камеры, посредством покомпонентного умножения исходного изображения на бинарную маску. Это обеспечивает «зануление» всех пикселей, относящихся к областям, исключенным из анализа. Подобный подход значительно снижает влияние на алгоритмы обработки изображений таких посторонних объектов, как элементы манипулятора или другое технологическое оборудование, находящееся в пределах рабочей зоны, и позволяет существенно сократить число ложных детекций, не прибегая к дополнительной фильтрации.

Следующим этапом обработки кадра является подсистема фонового вычитания, основанная на методе адаптивного

моделирования заднего плана сцен с использованием смеси гауссовых распределений. Для этого используется алгоритм MOG2 в реализации OpenCV, способный корректно обрабатывать сцены, освещенность которых изменяется медленно во времени [21]. Для каждого пикселя кадра MOG2 поддерживает взвешенную сумму нескольких гауссовых распределений, параметры которых обновляются с приходом каждого нового кадра. Пиксель принадлежит классу фона, если его текущее значение X_t находится в пределах $2.5\sigma_k$ от среднего значения μ_k одного из фоновых распределений:

$$|X_t - \mu_k| \leq 2.5\sigma_k,$$

где σ_k – стандартное отклонение от k -й гауссовой компоненты, характеризующее разброс значений около ее среднего μ_k . Пороговое значение на основе дисперсии в $2.5\sigma_k$ было подобрано эмпирически и примерно соответствует покрытию 98–99% данных нормального распределения, что обеспечивает баланс между чувствительностью и устойчивостью к шуму.

Если текущее значение пикселя не соответствует ни одному из известных фоновых распределений, алгоритм классифицирует его как часть переднего плана. Результатом работы алгоритма становится двухклассовая бинарная маска, отражающая текущее состояние кадра (рис. 2). Благодаря этому обеспечивается надежное выделение динамически появляющихся объектов на рабочей поверхности РТК при минимальной зависимости от колебаний освещенности и теневых эффектов.

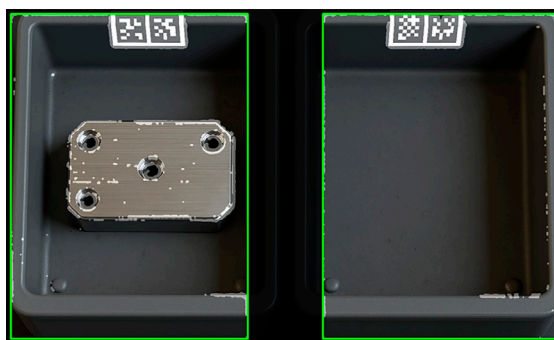


Рис. 2. Результат применения бинарной маски ROI и алгоритма MOG2 на кадре рабочей поверхности
Примечание: составлен авторами по результатам данного исследования

В дополнение к фоновому вычитанию применяется предобработка кадра, включающая конвертацию изображения в градации

серого, улучшение локального контраста методом адаптивной гистограммы с ограничением контраста CLAHE (Contrast Limited Adaptive Histogram Equalization) и сглаживание гауссовым фильтром. В типовой настройке использовались $\text{cliplimit} = 2,0$, $\text{tileGridSize} = 8 \times 8$, ядро сглаживания 5×5 . Процедура CLAHE вычисляет новую яркость пикселей по формуле

$$I'_{\text{CLAHE}}(x, y) = f_{\text{CLAHE}}(I(x, y); \Omega(x, y)),$$

где $I(x, y)$ – исходная яркость пикселя в точке (x, y) ;

$I'_{\text{CLAHE}}(x, y)$ – преобразованная яркость пикселя после применения CLAHE;

$\Omega(x, y)$ – локальная область вокруг пикселя (x, y) , используемая для построения гистограммы и адаптивной коррекции яркости.

После получения бинарной маски переднего плана следующим шагом является подсистема обнаружения AprilTag-кластеров, отвечающих за идентификацию посадочных позиций изделий. На предварительно подготовленную бинаризованную карту накладывают синхронное глубинное изображение, полученное стереокамерой. Для каждого кадра выполняется поиск заранее известных комбинаций идентификаторов AprilTag-меток, которые формируют стабильные геометрические кластеры. По результатам кластеризации каждая посадочная позиция описывается в виде прямоугольного ROI:

$$C_i = \{X_i, Y_i, Z_i, w_i, h_i, \theta_i\},$$

где X_i, Y_i, Z_i – координаты центра ячейки; w_i, h_i – размеры в плоскости стола, θ_i – ориентация.

Определение посадочных позиций деталей и инструментов становится независимым от их фактической геометрии и визуальных характеристик, что значительно упрощает переналадку и обеспечивает повторяемость результатов при изменении номенклатуры выпускаемой продукции.

Следующей в пайплайне является подсистема проверки заполненности каждой идентифицированной ячейки. Для C_i берется локальная область, заданная w_i, h_i, θ_i . Ячейка помечается как «занятая», если выполняется одно из двух условий:

$$\text{occupied}(C_i) = \begin{cases} 1, & \Delta z_i > z_{th}, \\ 1, & A_i > a_{th}, \\ 0, & \text{иначе} \end{cases},$$

где Δz_i – медианная высота над плоскостью дна внутри ROI, мм;

A_i – площадь переднего плана внутри ROI;

z_{th} – порог по высоте, мм;

a_{th} – порог по площади, px^2 .

Если хотя бы одна ячейка признана «занятой», в ее ROI запускается контурный детектор для уточнения контура детали. Вместо поиска произвольных контуров по всему кадру алгоритм работает с небольшим четко закрепленным набором «посадочных» точек, что существенно уменьшает ложные срабатывания и упрощает ввод новых типов деталей.

На рис. 3 представлена иллюстрация работы подсистемы обнаружения AprilTag-кластеров и проверки заполненности посадочных ячеек.

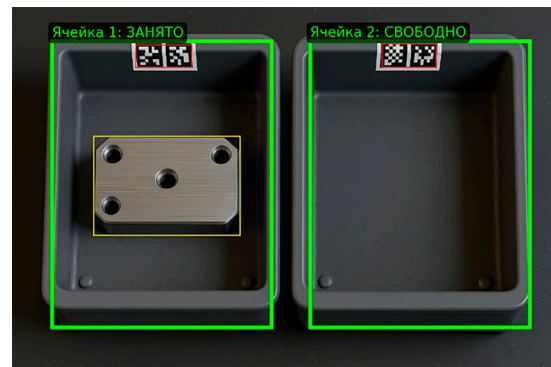


Рис. 3. Иллюстрация работы подсистемы обнаружения AprilTag-кластеров и проверки заполненности посадочных ячеек
Примечание: составлен авторами по результатам данного исследования

В предлагаемой методике рассматривается два варианта детектора – нейросетевой и контурный. В контурном режиме детекция происходит с использованием алгоритмов бинаризации и поиска контуров, формирующих ограничивающие рамки объектов. Контурный детектор работает в несколько последовательных этапов, обеспечивая эффективность и низкую вычислительную нагрузку при обработке изображений с простыми геометрическими формами. Первоначально изображение подвергается адаптивной пороговой обработке, благодаря чему достигается устойчивость к изменениям освещения и текстуры объекта. Затем на полученной бинарной маске выполняется поиск контуров методом поиска замкнутых границ, после чего осуществляется аппроксимация формы минимальным вписанным прямоугольником, который задает окончательные границы обнаруженного объекта. На рис. 4 представлена схема работы контурного детектора.

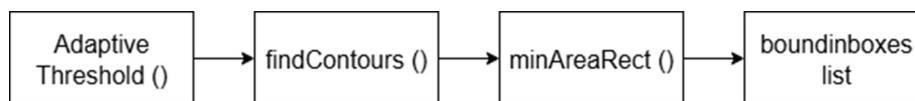


Рис. 4. Схема работы контурного детектора

Примечание: составлен авторами по результатам данного исследования

Если активирован нейросетевой режим, используется нейронная сеть YOLOv8-OBV. Стандартные параметры включали в себя разрешение 640x640, порог уверенности в пределах 0,25–0,35 и порог NMS по IoU 0,45. Нейросетевой детектор применяется для идентификации объектов со сложной геометрией в условиях нестабильного освещения. Нейросеть предварительно обучена на специально подготовленном наборе изображений посадочных ячеек и типовых заготовок, что обеспечивает точность и надежность распознавания объектов различных форм и размеров.

Для обучения YOLOv8-OBV использовался собственный датасет из 960 изображений ячеек, разделенный в пропорции 70%/15%/15% на обучающую, валидационную и тестовую выборки. Разметка выполнялась в формате ориентированных ограничивающих рамок для одного класса «заготовка», с привязкой к координатам ячейки. Для повышения обобщающей способности применялись аугментации, включавшие в себя: случайные повороты $\pm 15^\circ$, изменения яркости и контраста $\pm 20\%$, небольшие перспективные искажения (до 2%) и добавление шума. В качестве базовой архитектуры использовалась модель YOLOv8n-obv, обучение проводилось в течение

150 эпох с размером батча 16, оптимизатором Adam и начальной скоростью обучения $1 \cdot 10^{-4}$. По результатам обучения достигнут $mAP@.5 = 0,952$ на валидационной выборке, а кривые потерь обучения и валидации не показывали признаков переобучения.

Оба варианта возвращают список ориентированных ограничивающих рамок u, v, w, h, θ, Z , где u, v – центр рамки в пикселях, w, h – ее размеры, Z – медианная глубина внутри рамки, θ – ориентация. Для каждой рамки вычисляется 3D-центр в системе координат камеры, а затем в базовой системе координат робота по уже описанной схеме преобразований.

После детекции объектов возникает необходимость исключить повторную передачу координат объектов роботу-манипулятору, так как это может привести к нежелательным повторным действиям. Для предотвращения такой ситуации используется механизм подавления повторных детекций обнаруженных целей. На каждом шаге координаты вновь найденных объектов сопоставляются с массивом объектов, которые были опубликованы на предыдущих кадрах. Сравнение производится путем вычисления евклидова расстояния между текущими P_t и ранее переданными координатами предыдущего объекта P_{t-1} по следующей формуле:

$$P_t - P_{t-1} = \sqrt{(X_t - X_{t-1})^2 + (Y_t - Y_{t-1})^2 + (Z_t - Z_{t-1})^2} < d_{dup}.$$

Если вычисленное расстояние меньше установленного эмпирического порога d_{dup} (в мм), объект считается уже обработанным и его координаты повторно не отправляются роботу. В противном случае объект считается новым, его координаты передаются в последующие подсистемы метода. Буфер хранит последние $N = 5$ кадров, а для гашения дрожания введено «время удержания» $T_{hold} = 1$ с, в течение которого обслуженная цель остается заблокированной для повторной публикации. Значение d_{dup} в работе варьировалось от 10 до 30 мм в зависимости от размеров ячеек.

Следующий этап работы системы заключается в переводе пиксельных координат, полученных на этапе обнаружения объектов, в локальные метрические координаты рабочей зоны робота-манипуля-

тора. Этот этап является ключевым, так как от точности определения координат объектов зависит корректность последующих действий робота при манипуляциях с деталями и инструментами.

В предлагаемом методе используется hand-eye калибровка в конфигурации «камера на манипуляторе» с интеграцией библиотеки ROS2 и подсистемы преобразований координат TF2 [23]. TF2 представляет собой стандартный механизм ROS, который в реальном времени отслеживает и публикует преобразования между различными системами координат компонентов роботизированной системы. TF2 служит для публикации и композиции уже известных калибровочных преобразований в реальном времени, обеспечивая корректное применение трансформаций. Благодаря

этому обеспечивается динамическое и точное преобразование координат даже в условиях непрерывного движения и вибраций робота-манипулятора.

Первичная калибровка выполняется один раз и проходит в три этапа. Сначала настраиваются внутренние параметры и дисторсия камеры, после чего изображения ректифицируются с использованием стандартных алгоритмов библиотеки OpenCV. Далее выполняется hand-eye калибровка [5, 22] по классической схеме $AX = XB$ при наборе 20 поз манипулятора, равномерно покрывающих рабочий объем над столом. В качестве алгоритма использовался вариант Tsai-Lenz, реализованный в среде ROS2. Качество калибровки оценивалось по среднеквадратической репроекционной ошибке и 3D-смещению метки. Среднее значение составило 0,45 пикселя по изображению, что соответствует примерно 0,6 мм на рабочей дистанции 0,7 м. В результате определялось однородное преобразование от камеры к фланцу инструмента робота $T_{tool \leftarrow cam}$.

На третьем этапе задается положение рабочей поверхности в базовой системе координат робота $T_{base \leftarrow table}$ по кластеру AprilTag и оцененной гомографии плоскости стола. На этом этапе система TF2 повторную калибровку не проводит, а в реальном времени композитирует актуальные преобразования по текущей кинематике робота и публикует:

$$T_{base \leftarrow cam} = T_{base \leftarrow tool} \cdot T_{tool \leftarrow cam},$$

где $T_{base \leftarrow tool}$ берется из текущих углов звеньев. Повторная hand-eye калибровка в ходе описанных экспериментов требуется только при замене камеры или изменениях ее механического крепления. Положение стола относительно базовой системы робота контролируется по контрольным кадрам с AprilTag один раз в начале каждой серии измерений.

После завершения этапа уточняющей детекции каждый объект описывается центром в изображении (u, v) и глубиной Z_{cam} . Эти пиксельные координаты необходимо выразить в базовой системе координат робота. Для этого используют два последовательно применяемых преобразования [24–26].

Сначала происходит преобразование пикселя к камерным 3D-координатам с учетом внутренних параметров камеры и предварительной компенсации дисторсии. Координаты точки в камерной системе вычисляются по однородной модели проекции:

$$P_{cam} = Z_{cam} K^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} (u - c_x) Z_{cam} / f_x \\ (v - c_y) Z_{cam} / f_y \\ Z_{cam} \end{bmatrix},$$

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}.$$

На втором шаге осуществляется преобразование из системы координат камеры в базовую систему координат робота. Для этого в ROS2 используется подсистема TF2, которая непрерывно публикует актуальную однородную матрицу перехода:

$$T_{base \leftarrow cam}(t) = \begin{bmatrix} R(t) & t(t) \\ 0^T & 1 \end{bmatrix},$$

$$R \in R^{3 \times 3}, \quad t \in R^3,$$

где R – текущая ориентация камеры, t – ее смещение относительно системы координат робота.

Итоговые координаты центра, ориентированного ROI, вычисляются как

$$\begin{bmatrix} P_{base} \\ 1 \end{bmatrix} T_{base \leftarrow cam}(t) \begin{bmatrix} P_{cam} \\ 1 \end{bmatrix}.$$

Пиксельные координаты со встроенной глубиной переводятся в рабочую систему робота, а ошибки, связанные с наклоном камеры, перспективой и колебаниями мобильного комплекса, компенсируются механизмом TF2. Эти 3D-координаты вместе с углом ориентации θ передаются в управляющий ROS-топик, на чем геометрический цикл обработки завершается.

Пример листинга реализации описанного преобразования координат с использованием TF2 в ROS2 представлен ниже (листинг 1).

В рабочей реализации данный фрагмент включен в состав узла ROS2 на языке Python, который инициализирует клиентскую библиотеку, создает объект узла и обрабатывает возможные исключения TF2. Эти элементы несущественны для понимания математической сути преобразования и опущены ввиду объемных листингов.

Для оценки эффективности разработанного метода были проведены испытания системы компьютерного зрения, направленные на сравнение контурного и нейросетевого детекторов. Ключевым элементом испытаний стала проверка точности и стабильности определения координат посадочных ячеек по группам AprilTag-меток, поскольку они являются фундаментом всей системы позиционирования.

```

from geometry_msgs.msg import PointStamped
from tf2_ros import Buffer, TransformListener
from tf2_geometry_msgs.tf2_geometry_msgs import do_transform_point
# Буфер TF2 и Listener создаются один раз в узле ROS2
tf_buffer = Buffer() # буфер для хранения трансформаций между системами координат
tf_listener = TransformListener(tf_buffer, node) # listener пополняет буфер

def cam_to_base(x_cam: float, y_cam: float, z_cam: float) -> PointStamped:
    «»»Преобразует точку из СК камеры cam в базовую СК робота base.«»»
    p_cam = PointStamped() #создаём сообщение с точкой в системе координат камеры
    p_cam.header.stamp = node.get_clock().now().to_msg() # штамп времени кадра
    p_cam.header.frame_id = 'cam' # точка задана в системе координат камеры 'cam'
    p_cam.point.x = x_cam # координата X точки в системе камеры
    p_cam.point.y = y_cam # координата Y точки в системе камеры
    p_cam.point.z = z_cam # координата Z точки в системе камеры
    # Берём актуальное преобразование base <- cam
    t = tf_buffer.lookup_transform(
        target_frame='base', # Целевая система координат - базовая система робота 'base'
        source_frame='cam', # исходная система координат - камера 'cam'
        time=rospy.time.Time() # Используем последнее доступное преобразование
    )
    p_base = do_transform_point(p_cam, t) # трансформация к точке и получение
    координат в СК 'base'
    return p_base # точка выраженная в базовой системе координат робота

```

Листинг 1. Фрагмент кода преобразования точки из системы координат камеры cam в базовую систему робота base с использованием TF2 на языке Python

Примечание: составлен авторами по результатам данного исследования

В испытаниях были задействованы робот-манипулятор Universal Robots UR5, стереокамера Intel RealSense D455 со стереобазой 95 мм, 1280×720@30 к/с. Использованы метки AprilTag tag36h11, посадочные ячейки представлены в виде физических лотков размера 100×120 мм, высота борта 20 мм. Семейство AprilTag tag36h11 выбрано как компромисс между помехоустойчивостью и габаритами маркировки. Код tag36h11 обладает большим алфавитом идентификаторов и высоким расстоянием Хэмминга между кодовыми словами, что снижает вероятность ложных срабатываний и ошибок декодирования при засветах и частичных окклюзиях. По сравнению с типовыми ArUco-семействами это дает более надежную детекцию на блестящих и слаботекстурных поверхностях, а по сравнению с ChArUco-таблицами не требует установки крупных шахматных мишеней, что важно в условиях ограниченных размеров ячеек.

Экспериментальная серия была организована в виде трех сценариев, отражающих типичные условия роботизированного рабочего места:

1. Простая геометрия (втулки, цилиндры) и стабильное освещение (600–700 лк), блики < 5 %, окклюзия меток 0 % – детали стандартной формы с равномерным освещением и минимальными вариациями в позиционировании посадочных мест.

2. Сложная геометрия (ступенчатые валы) и стабильное освещение (600–700 лк), блики 5–10 %, окклюзия меток < 25 % – детали сложной формы с большим количеством мелких элементов и изменяющимися расположениями посадочных ячеек.

3. Сложная геометрия и нестабильное освещение (120–350 лк), блики 10–20 %, окклюзия меток < 40 % – детали сложной формы с изменяющимися условиями освещения и различным взаимным расположением посадочных ячеек.

Оценка производилась по следующим показателям:

1. Ошибка AprilTag – средняя евклидова ошибка центра кластера, мм.

2. Ошибка объекта – средняя евклидова ошибка 3D-центра заготовки, мм.

3. Точность и полнота детекции (precision и recall) объектов.

4. Среднее время обработки одного кадра от момента захвата изображения до публикации координат в ROS2 (мс).

Эталонные координаты центров рабочих позиций и заготовок задавались в базовой системе робота путем обучения контрольных точек на UR5 с помощью штатного пульта ручного управления. Для центра каждой ячейки и для центра каждой тестовой заготовки выбиралось касательное положение захвата, после чего регистрировались координаты TCP в системе base.

Согласно паспорту UR5 и многократным повторным подходам, разброс этих измерений не превышал 0,2–0,3 мм, что задает верхнюю оценку погрешности эталонной разметки.

Освещенность в зоне ячеек контролировалась цифровым люксметром, замеры выполнялись в центре рабочей области на высоте 150 мм над плоскостью столов. Для сценариев с нестабильным освещением уровень освещенности изменялся за счет регулировки мощности и направления светильников, при этом интервалы фиксировались по показаниям люксметра. Процент окклюзии меток задавался как отношение суммарной площади закрытых частей AprilTag к полной площади всех меток в кластере. Окклюзия моделировалась непрозрачными масками, наклеенными на кромку ячейки. Блики формировались направленным светодиодным прожектором, а их доля оценивалась по бинаризованной маске пересвеченных пикселей как процент от площади ROI.

Каждый сценарий испытаний проводился по 50 раз, после чего результаты усреднялись. В качестве базового метода сравнения использовалась ручная экспертная разметка координат посадочных мест и заготовок. Результаты вычислительных экспериментов, проведенных на базе лаборатории «Умные производственные системы» РТУ МИРЭА, приведены в таблице.

При профилировании стендовой реализации было установлено, что основное время обработки кадра распределялось между тремя подсистемами: фоновым вычитанием MOG2, детекцией AprilTag и уточняющей детекцией внутри ячеек. Для контурного детектора суммарное время одного цикла составляло в среднем 36 мс (сценарий 1), из которых порядка 8–10 мс приходилось на обновление фоновой модели и морфологическую фильтрацию, 6–8 мс – на детекцию и декодирование AprilTag и 10–12 мс – на локальную бинаризацию и обработку

контуров в пределах ячеек, остальное – расходы ROS2/TF2. При использовании нейросетевого детектора YOLOv8-OBV общее время увеличивалось до 60–65 мс за счет вычислительно затратного прохода модели по ограниченным областям интереса, при этом вклад фонового вычитания и AprilTag оставался на том же уровне. Масштабируемость по числу ячеек определялась в первую очередь этапом детекции меток. Вычислительная сложность возрастала пропорционально числу меток в поле зрения. При типичной конфигурации (2 метки на ячейку) увеличение числа ячеек с 4 до 8 приводило к росту времени AprilTag-детекции примерно в 1,4–1,5 раза, но не изменяло затрат MOG2 и CLANE, которые зависят только от размера кадра. Нагрузка от уточняющей детекции возрастала с числом «занятых» ячеек. В контурном режиме обработка дополнительных ROI показала почти линейный рост времени, а в нейросетевом режиме количество активных ячеек непосредственно определяло число проходов YOLOv8-OBV.

Для оценки статистической значимости различий между детекторами дополнительно проводилась проверка распределений ошибок на нормальность по критерию Шапиро – Уилка и парное сравнение средних значений по t-критерию для связанных выборок. Во всех сценариях распределение евклидовой ошибки локализации не противоречило нормальному ($p > 0,05$), а разница между контурным и нейросетевым детектором по ошибке объекта в сценариях 2 и 3 оказалась статистически значимой ($p < 0,01$). Выбросы с величиной ошибки более 3σ встречались реже 2% наблюдений и в анализе не исключались. Для основных метрик (ошибка AprilTag и ошибка объекта) 95-процентные доверительные интервалы не выходили за пределы среднего $\pm 2\sigma$, что согласуется с принятым способом представления «среднее \pm стандартное отклонение».

Результаты экспериментов по работе метода в различных сценариях

№ сценария	Метод детекции	Ошибка AprilTag, мм	Ошибка объекта, мм	Precision, %	Recall, %	Время обработки, мс
1	Контурный детектор	$0,5 \pm 0,2$	$0,9 \pm 0,4$	97,6	96,8	36 ± 6
1	YOLOv8-OBV	$0,5 \pm 0,2$	$1,0 \pm 0,5$	96,5	97,0	59 ± 7
2	Контурный детектор	$0,9 \pm 0,4$	$3,5 \pm 1,1$	84,3	81,2	39 ± 6
2	YOLOv8-OBV	$0,9 \pm 0,4$	$1,6 \pm 0,6$	95,7	94,6	60 ± 8
3	Контурный детектор	$1,3 \pm 0,6$	$7,0 \pm 1,3$	68,5	65,3	47 ± 8
3	YOLOv8-OBV	$1,2 \pm 0,5$	$2,5 \pm 0,7$	90,2	88,7	64 ± 9

Примечание: составлена авторами на основе полученных данных в ходе исследования.

Отдельно была рассмотрена ожидаемая теоретическая точность локализации центров ячеек. Погрешность координат центра кластера AprilTag оценивалась суммой вклада ошибки 2D-локализации вершин меток и погрешности по глубине стереокамеры. При рабочем диапазоне расстояний 0,6–0,8 м, фокусном расстоянии порядка 850 пикселей и типичной точности субпиксельной локализации уголков меток 0,1–0,2 пикселя погрешность в плоскости стола не превышала 0,1–0,2 мм. Для камеры RealSense D455 при базисе 95 мм дисперсия глубины составляла порядка 1–1,5 мм. Суммарная среднеквадратичная ошибка центра кластера оценивалась как

$$\sigma_{cell} \approx \sqrt{\sigma_{tag}^2 + \sigma_z^2} \leq 1,3 - 1,5 \text{ мм},$$

что согласуется с экспериментальными значениями ошибки AprilTag в таблице.

В условиях простой геометрии и стабильного освещения обе версии детекторов показали сопоставимую высокую точность определения как посадочных ячеек по AprilTag-меткам, так и заготовок. При этом контурный метод показал преимущество по скорости обработки, что объясняется его меньшей вычислительной сложностью.

При увеличении сложности геометрии и вариативности расположения посадочных ячеек точность детекции по меткам AprilTag оставалась высокой (до 1 мм), но погрешность определения координат заготовок у контурного детектора значительно возросла, тогда как нейросетевой детектор YOLOv8-OBV сохранил стабильно низкие значения погрешности.

В условиях нестабильного освещения обе версии системы показали некоторое ухудшение точности определения координат посадочных меток, но нейросетевой детектор продемонстрировал значительно меньшую погрешность распознавания заготовок, что подчеркивает его устойчивость к условиям изменяющегося освещения.

По результатам испытаний можно сформулировать следующие рабочие ограничения метода:

1. Освещенность не ниже 300 лк для сохранения ошибки AprilTag $\leq 1,3$ мм.
2. Допустимая окклюзия меток до 40 % площади кластера.
3. Доля бликов до 10–20 % в ROI, при превышении чего контурный детектор деградирует и рекомендуется переход к YOLOv8-OBV.
4. Hand-eye калибровка и привязка стола требуют повторной проверки только при механическом вмешательстве.

Как итог, кластеры AprilTag обеспечивают стабильную «якорную» привязку ($\leq 1,3$ мм даже при сложном сценарии). Выбор детектора рационально делать по условиям: контурный – при простой геометрии и стабильном свете обеспечивает минимальную задержку, YOLOv8-OBV – при сложной форме или нестабильной освещенности. Частота публикаций поз составляет примерно 28 Гц (контурный) и 16 Гц (YOLOv8-OBV), что позволяет подбирать режим под требования такта операции.

Заключение

Полученные результаты экспериментальных испытаний подтверждают эффективность и целесообразность предложенного метода компьютерного зрения для роботизированных рабочих мест мелкосерийного производства с часто меняющейся номенклатурой изделий. Использование AprilTag-меток в качестве стабильных метрических ориентиров обеспечивает высокую точность и надежность первоначального определения координат посадочных ячеек. Реализованная возможность выбора между контурным и нейросетевым детекторами позволяет гибко адаптировать систему под различные производственные условия, обеспечивая баланс между скоростью и точностью обработки. Предложенный подход позволяет существенно сократить количество ложных срабатываний, повысить оперативность и точность переналадки роботаманипулятора в динамически изменяющихся условиях мелкосерийного производства.

Авторы отмечают и ограничения предлагаемой методики, в которые входят: необходимость установки и обслуживания меток, достаточные условия освещенности, необходимость первоначальной калибровки и поддержка корректной TF-иерархии. Особенностью представленных результатов является то, что все эксперименты проводились на одной модели робота и одной камере на стационарном стенде. Влияние отличающейся кинематики, других типов стереокамер и вибраций мобильной платформы специально не исследовалось. Тем не менее предлагаемая методика опирается на стандартный стек ROS2/TF2 и hand-eye калибровку, что делает ее переносимой на другие конфигурации. Экспериментальная проверка на мобильных манипуляторах планируется в последующих работах.

С точки зрения практической эффективности основной эффект от внедрения метода проявляется в сокращении инженерных трудозатрат на повторную настройку системы машинного зрения. Переход от сценориентированной калибровки и переоб-

учения детекторов к привязке меток ячеек позволяет при смене номенклатуры ограничиться механической перестановкой ячеек и обновлением конфигурационного файла без привлечения специалистов по компьютерному зрению. В условиях мелкосерийного производства с частыми переналадками это снижает затраты времени на ввод новой партии в работу и уменьшает риск простоев из-за ошибок перенастройки.

Методика целесообразна для задач комплектования, обслуживания станков, быстрой смены оснастки в ячейках с изменяемой номенклатурой, а также для мобильных комплексов, в которых важно быстро восстанавливать привязку стола после его перемещения. Направления дальнейших исследований будут посвящены адаптивной настройке порогов z_{th} и a_{th} с автоматическим дрейф-контролем под изменения сцены, а также совершенствованию метода при частичной/полной потере части меток, включая трекинг кластеров, предсказание гомографии по прошлым наблюдениям и мультикамерное слияние.

Список литературы

1. Burde V., Martinez D., Zeman V. et al. Automatic Workspace Calibration Using Homography for Pick and Place // *IEEE CASE*. 2023. Vol. 1–6. DOI: 10.1109/CASE56687.2023.10260601.
2. Nguyen H.N., Le P.N., Kang H.J. A new calibration method for enhancing robot position accuracy by combining a robot model identification and an ANN based error compensation technique // *Advances in Mechanical Engineering*. 2019. Vol. 11. Is. 1. P. 1–11. DOI: 10.1177/1687814018822935.
3. Allegro D., Terreran M., Ghidoni S. Multi-Camera Hand-Eye Calibration for Human-Robot Collaboration in Industrial Robotic Workcells // *IEEE Robotics and Automation Letters*. November 2024. PP (99). P. 1–8. DOI: 10.1109/LRA.2024.3468089.
4. Jiang Z., Zhou W., Li H. et al. A New Kind of Accurate Calibration Method for Robotic Kinematic Parameters Based on Extended Kalman and Particle Filter Algorithm // *IEEE Transactions on Industrial Electronics*. 2018. Vol. 65. Is. 4. P. 3337–3345. DOI: 10.1109/TIE.2017.2748058.
5. Enebusi I., Foo M., Ibrahim B.S.K.K. et al. A Comparative Review of Hand-Eye Calibration Techniques for Vision-Guided Robots // *IEEE Access*. 2021. Vol. 9. P. 113143–113155. DOI: 10.1109/ACCESS.2021.3104514.
6. Wang J., Li L., Xu P. Visual Sensing and Depth Perception for Welding Robots and Their Industrial Applications // *Sensors*. 2023. Vol. 23. Is. 24. P. 9700. DOI: 10.3390/s23249700.
7. Li M., Huang J., Xue L., Zhang R. A guidance system for robotic welding based on an improved YOLOv5 algorithm with a RealSense depth camera // *Scientific Reports*. 2023. Vol. 13. DOI: 10.1038/s41598-023-48318-8.
8. Velickaitė G., Daugėla I., Suzdalev I. Unsupervised Binary Classifier-Based Object Detection Algorithm with Integrated Background Subtraction Suitable for Use with Aerial Imagery // *Applied Sciences*. 2025. Vol. 15. P. 8608. DOI: 10.3390/app15158608.
9. Zhou Y. et al. Robust Task-Oriented Markerless Extrinsic Calibration for Robotic Pick-and-Place Scenarios // *IEEE Access*. 2019. Vol. 7. P. 127932–127942. DOI: 10.1109/ACCESS.2019.2913421.
10. Егорцев М.В., Диане С.К., Кац Н.Д. Алгоритмическое обеспечение системы внешнего наблюдения и маршрутизации автономных мобильных роботов // *Российский технологический журнал*. 2021. Т. 9. № 3. С. 15–23. DOI: 10.32362/2500-316X-2021-9-3-15-23.
11. Ibrahim A.A.M.S., Tapamo J.-R. A Survey of Vision-Based Methods for Surface Defects' Detection and Classification in Steel Products // *Informatics*. 2024. Vol. 11. P. 25. DOI: 10.3390/informatics11020025.
12. Kimble K., Van Wyk K., Falco J. et al. Benchmarking Protocols for Evaluating Small Parts Robotic Assembly Systems // *IEEE Robotics and Automation Letters*. 2020. Vol. 5. Is. 2. P. 883–890. DOI: 10.1109/LRA.2020.2965869.
13. Tsai C.-H., Lin J.-J., Hsieh T.-F., Yen J.-Y. Trajectory Control of An Articulated Robot Based on Direct Reinforcement Learning // *Robotics*. 2022. Vol. 11. Is. 5. P. 116. DOI: 10.3390/robotics11050116.
14. Boschetti G., Sinico T., Trevisani A. Improving Robotic Bin-Picking Performances through Human-Robot Collaboration // *Applied Sciences*. 2023. Vol. 13. P. 5429. DOI: 10.3390/app13095429.
15. Ojer M., Lin X., Tammara A., Sanchez J. R. PickingDK: A Framework for Industrial Bin-Picking Applications // *Applied Sciences*. 2022. Vol. 12. P. 9200. DOI: 10.3390/app12189200.
16. Wang Z., Zhang Z., Zhu W., Hu X., Deng H., He G., Kang X. A Robust Planar Marker-Based Visual SLAM // *Sensors*. 2023. Vol. 23. Is. 2. P. 917. DOI: 10.3390/s23020917.
17. Westman A. et al. High-Precision Pose Estimation of Industrial Robots Using AprilTag Observations // *Technical Report CMU-RI-TR-18-43*. Carnegie Mellon University, 2018. [Электронный ресурс]. URL: <https://www.ri.cmu.edu/publications/underwater-apriltag-slam-and-calibration-for-high-precision-robot-localization/> (дата обращения: 10.07.2025).
18. Chen A., Li X., Cen K., Hon C. Adaptive Grasp Pose Optimization for Robotic Arms Using Low-Cost Depth Sensors in Complex Environments // *Sensors*. 2025. Vol. 25. Is. 3. P. 909. DOI: 10.3390/s25030909.
19. Guo N., Zhang B., Zhou J. et al. Pose estimation and adaptable grasp configuration with point cloud registration and geometry understanding for fruit grasp planning // *Computers and Electronics in Agriculture*. 2020. Vol. 179. Art. 105818. DOI: 10.1016/j.compag.2020.105818.
20. Холопов В.А., Макаров М.А., Благовещенский И.Г. Метод согласования систем координат компонентов в РТК на основе машинного зрения // *Информационные и математические технологии в науке и управлении*. 2024. № 4 (36). С. 26–36. DOI: 10.25729/ESI.2024.36.4.003.
21. Zivkovic Z. Improved adaptive Gaussian mixture model for background subtraction // *Proceedings of the 17th International Conference on Pattern Recognition (ICPR 2004)*. 2004. Vol. 2. P. 28–31. DOI: 10.1109/ICPR.2004.1333992.
22. Song Z., Sun C., Sun Y., Qi L. Robotic Hand-Eye Calibration Method Using Arbitrary Targets Based on Refined Two-Step Registration // *Sensors*. 2025. Vol. 25. P. 2976. DOI: 10.3390/s25102976.
23. Холопов В.А., Макаров М.А., Кладов А.С. Метод определения траектории робота-манипулятора в роботизированных технологических комплексах с учетом временных ограничений // *Современные наукоемкие технологии*. 2024. № 12. С. 113–121. DOI: 10.17513/snt.40251.
24. Do Q.T., Chang W.Y., Chen L.W. Dynamic Workpiece Modeling with Robotic Pick-Place Based on Stereo Vision Scanning Using Fast Point Feature Histogram Algorithm // *Applied Sciences*. 2021. Vol. 11. Is. 24. P. 11522. DOI: 10.3390/app112311522.
25. Ruoppa L., Oinonen O., Taher J., Lehtomäki M., Takhtkeshha N., Kukko A., Kaartinen H., Hyypä J. Unsupervised deep learning for semantic segmentation of multispectral LiDAR forest point clouds // *ISPRS Journal of Photogrammetry and Remote Sensing*. 2025. Vol. 228. P. 694–722. DOI: 10.1016/j.isprsjsr.2025.07.038.
26. Юрманов А.П., Панчук М.О., Коноплин А.Ю. Метод коррекции траекторий рабочего органа многозвенного манипулятора необитаемого подводного аппарата // *Подводные исследования и робототехника*. 2023. № 4 (46). С. 43–51. DOI: 10.37102/1992-4429_2023_46_04_04. EDN: EIPJUC.

Конфликт интересов: Авторы заявляют об отсутствии конфликта интересов.

Conflict of interest: The authors declare that there is no conflict of interest.