На правах рукописи

Староверов Алексей Витальевич

Иерархические методы и алгоритмы визуальной навигации внутри помещений с обучаемыми навыками

Специальность 1.2.2 —
«Математическое моделирование, численные методы и комплексы программ»

Автореферат

диссертации на соискание учёной степени
кандидата физико-математических наук

Долгопрудный — 2023

Общая характеристика работы

Актуальность темы исследования.

В последнее время у научного сообщества появился большой интерес к задачам воплощенного искусственного интеллекта (ВИИ/ Embodied AI) [7—10]. Особенность их заключается во взаимодействии агента (робота) с окружающими объектами в человеко-ориентированных средах. Одной из главной задачей ВИИ является способность агента свободно ориентироваться в новых для него средах и оперировать семантическими априорными знаниями на основе прошлого опыта [11]. Для этого мобильный робот должен уметь решать различные подзадачи, или, иными словами, обладать разными навыками, такими как: обнаружение и сегментирование объектов внешней среды [12—14], локализация и картирования (SLAM, [15; 16]), навигация [17], планирование [18] и т.д. На практике, эти навыки объединяются в единую систему, где каждый модуль отвечает за свою подзадачу и может быть реализован как на основе нейронных сетей, так и классическим способом [19; 20]. Альтернативой данного подхода являются полностью самообучаемые системы [21; 22], которые могут быть сформулированы, как связанные с обучением с подкреплением [23]. Агент учится выполнять оптимальные последовательности действий для выполнения задачи с максимальной совокупной наградой, получая обратную связь от окружающей среды. При этом агент не получает прямых инструкций для решения задачи. Самообучающиеся системы, в которых используются агенты, основанные на обучении с подкреплением, достигли впечатляющих результатов во все более сложных областях [8; 21; 24; 25].

Одним из перспективных подходов для объединения навыков агента в единую архитектуру является иерархическое обучение с подкреплением (HRL) [26]. Иерархический подход позволяет разделить сложную задачу на множество подзадач. Для людей это является естественной процедурой. Однако, остается до конца не изученным вопрос – как именно человеку удается находить соответствующую иерархическую структуру. Поиск хорошей декомпозиции на подзадачи часто творческая задача, решение которой представляет серьезную проблему. Несмотря на то что в этом направлении получен ряд достижений [27], автоматическое построение иерархической структуры остается открытой проблемой в обучении с подкреплением. Методы HRL позво-

ляют агентам разложить задачу на более простые подзадачи. HRL-подходы обучают агентов различным уровням стратегии, каждый из которых специализируется на принятии решений в различных временных масштабах.

Для успешного применения обучаемых алгоритмов, однако, требуется симуляционная среда, которая будет обеспечивать большие объемы данных и симулировать все внешние условия, под которые агент будет приспосабливаться. Особенно важно это при переносе агента из симулятора в реальность. Если в реальности условия будут сильно отличаться, то агент не будет способен их преодолеть, а обучение в реальном мире занимает непропорционально много времени и может быть небезопасным, так как на первых шагах агент будет предпринимать случайные действия, перед тем как выучить приемлемую стратегию поведения. Это мотивировало созданию таких симуляционных сред как Habitat [7] и BPS [28], которые будут использованы далее в работе.

В данных средах за последнее время было продемонстрировано много успешных алгоритмов, которые применяя модульные [20; 29; 1; 2] или иерархические подходы [29; 30] превосходили классические подходы [21], что послужило убедительным доказательством того, что системы искусственного интеллекта могут быть масштабированы для работы в сложных, динамичных средах и применяться на реальных робототехнических системах. Однако как показывает опыт соревнований Habitat Challenge [7], современные методы недостаточно хорошо справляются с задачами, где требуется семантическое понимание сцены, успешно завершая эпизоды только в половине случаев.

В следствии этого была выбрана тема диссертационной работы.

Целью данной работы является повышение автономности робототехнических систем в задаче навигации на основе разработки гибридных методов визуальной навигации с использованием обучаемых и необучаемых навыков с возможностью использования на реальном роботе.

Для достижения поставленной цели были определены и решены следующие **задачи:**

– Разработать иерархический обучаемый метод решения задачи навигации в 2D и визуальных 3D средах. Интегрировать семантическую сегментацию, картирование и локализацию в обучаемый метод поиска целевых объектов.

– Для задачи навигации к семантическим объектам разработать метод визуальной навигации с использованием минимально необходимых априорных знаний о структуре среды. Выделить в поведении агента предварительно обученные стратегии поведения, которые можно объединить и повторно использовать в различных навигационных задачах без каких-либо изменений. Исследовать методы 3D реконструкции сцен и возможность использования их как симулятора для дообучения навыков агента для применения на реальном мобильном роботе.
– Разработать гибридный алгоритм решения задачи навигации, объединяющий классические и обучаемые навыки агента с обучаемым модулем переключения стратегий. Адаптировать предложенный метод под реального мобильного робота в зашумленных условиях.

Научная новизна:

– Был предложен метод интеграции семантической сегментации, картирования, локализации и обучения с подкреплением для повышения эффективности исследования окружающей среды, поиска целевого объекта и быстрой навигации к нему. Для задачи навигации до точки был предложен иерархический метод с выделением подцелей. Особенностью данного метода является одновременное обучение всех уровней иерархий в условиях разряженной награды от среды.
– Задача поиска целевых объектов на карте была сформулирована через навыки агента и предложен вариант использования опорных областей для ускорения исследования сцен для мобильного робота в человеко-ориентированных помещениях. Был представлен метод иерархической стратегии с ориентирами HLPO, который использует доступную информацию о ориентирах и на основе нее выстраивает иерархию из заранее обученных навыков агента, что улучшает способность агента исследовать среду в два раза. Полученный метод был перенесен на реального робота путем дообучения стратегии в реконструированной среде реального помещения.
– Выполнено оригинальное исследование, в рамках которого был разработан объединяющий классические и на основе обучения с под-

креплением навыки агента алгоритм SkillFusion, показавший в задаче навигации к целевым объектам свое преимущество перед только классическими или обучаемыми стратегиями. Выбор навыков осуществляется на основе модуля оценки их полезности в каждый момент времени.

Теоретическая значимость работы заключается в следующем:

– Предложен метод, совмещающий классические алгоритмы планирования и методы обучения с подкреплением. Его главная особенность заключается в том, что он учитывает преимущества обоих парадигм и динамически выбирает в зависимости от состояния агента и оценки функции полезности каждого навыка, какой навык использовать в текущий момент.
– Предложен новый подход к решению задачи навигации поиска целевых объектов с использованием ориентиров. С обновленной формулировкой задачи была создана новая иерархическая архитектура, в которой используются навыки, которые можно комбинировать и повторно использовать в различных навигационных задачах без изменений.

Практическая значимость работы заключается в следующем:

– Для методов семантической сегментации, картирования и локализации был собран и выложен в открытый доступ оригинальный набор данных HISNav [3].
– Отработан метод 3D реконструкции реального помещения и использование его в симуляторе для обучения алгоритмов на основе обучения с подкреплением.
– Предложенный способ использования методов обучения с подкреплением для задач навигации был испытан на мобильных роботах в реальных условиях и легко адаптируется на разные робототехнические платформы. В будущем данный подход может быть расширен на семантически более сложные постановки задачи, тем самым повышая степень автономности робототехнических систем.

Методология и методы исследования.

Разрабатываемые алгоритмы основываются на методах машинного обучения, теории графов, методах оптимизации и статистике. Основным ме-

тодом оценки эффективности предложенных результатов в данном исследовании является численный эксперимент. Сравнительный анализ эффективности алгоритмов проводится на основе статистического анализа нескольких запусков каждого из алгоритмов. В дополнение к этому, в работе проводится абляционное исследование, которое позволяет оценить вклад отдельных элементов дизайна нового предложенного решения в конечный результат. Реализация всех рассматриваемых алгоритмов и экспериментов осуществлена с использованием языков программирования Python3 и bash, а также дополнительных технологий, таких как библиотека машинного обучения PyTorch, программа для контейнеризации приложений docker, библиотека numpy и другие.

Положения, выносимые на публичное представление:

- Обучаемый метод выделения подцелей с интеграцией метода SLAM и семантической сегментации для задачи навигации до точки и поиска заданных объектов.
- Оригинальный алгоритм визуальной навигации с использованием опорных областей для ускорения исследования сцен для мобильного робота в человеко-ориентированных средах.
- Гибридный метод, объединяющий классические и обучаемые подходы для решения задачи поиска целевых объектов на основе функции полезности каждого навыка. Данный метод занял первое место на международном соревновании по навигации Habitat Challenge 2023 [1]

Достоверность результатов, полученных в ходе исследования, обеспечивается использованием методики численного эксперимента. Представленные алгоритмы описаны подробно, что позволяет повторить их результаты. Для каждого алгоритма представлено детальное описание и код выложен в открытый доступ. Многие из полученных данных согласуются и дополняют результаты, полученные в работах других исследователей.

Апробация работы. Основные результаты работы докладывались на:

- XXI Международная научно-техническая конференция “Нейроинформатика-2019”, 7-11 октября 2019, Москва

[1] https://aihabitat.org/challenge/2023/

– XXII Международная научно-техническая конференция “Нейроинформатика-2020”, 12-16 октября 2020, Москва
– VI Всероссийский научно-практический семинар “Беспилотные транспортные средства с элементами искусственного интеллекта” (БТС-ИИ 2021), 16-19 ноября 2021, Москва
– Научно-практический семинар Центра когнитивного моделирования ФПМИ МФТИ, 11 мая 2023, Москва
– IEEE/CVF Conference on Computer Vision and Pattern Recognition 2023, “CVPR-2023”, Embodied AI Workshop, 18-22 июня, Ванкувер

Содержание диссертации соответствует паспорту специальности 1.2.2. Математическое моделирование, численные методы и комплексы программ, в частности, пунктам:

– п. 2 – Разработка, обоснование и тестирование эффективных вычислительных методов с применением современных компьютерных технологий.
– п. 3 – Реализация эффективных численных методов и алгоритмов в виде комплексов проблемно-ориентированных программ для проведения вычислительного эксперимента.
– п. 5 – Разработка новых математических методов и алгоритмов валидации математических моделей объектов на основе данных натурного эксперимента или на основе анализа математических моделей.
– п. 8 – Комплексные исследования научных и технических проблем с применением современной технологии математического моделирования и вычислительного эксперимента.

Личный вклад. В работе [3] – разработка метода интеграции семантической сегментации, картирования, локализации и обучения с подкреплением для задачи навигации и иерархического метода выделения подцелей; В работе [4] автор предложил постановку задачи навигации с ориентирами, разработал метод иерархической стратегии с ориентирами и метод переноса обученной стартегии на реального робота. В работе [5] автор предложил и реализовал метод интеграции классических и обучаемых навыков.

Публикации. Основные результаты по теме диссертации изложены в 3 печатных изданиях [3—5], 3 в периодических научных журналах, индекси-

руемых Scopus, в том числе 3 из которых опубликованы в журналах первого квартиля.

Основное содержание

Во **введении** данной диссертационной работы подчеркивается актуальность проводимых исследований, что подтверждается обзором научной литературы, связанной с изучаемой проблематикой. Далее формулируется основная цель исследования, определяются задачи работы, а также раскрывается научная новизна и практическая значимость представляемого исследования.

Первая глава посвящена обзору методов решения задачи визуальной навигации. В начале главы определяется постановка задачи навигации и описываются особенности задачи навигации до координат и до целевых объектов.

После описывается симуляционная среда Habitat [7], используемая в дальнейших экспериментах в данной работе. В ней есть все необходимые компоненты для обучения методов с подкреплением и возможна реализация методов SLAM и алгоритмов классического планирования. Для обеспечения визуального сходства симуляции с реальностью, необходимы снимки 3D реконструированных реальных помещений. Среди многих наборов данных для этой цели был выбран Habitat-Matterport 3D (HM3D [31]), он является самым обширным и «парето-оптимальным» в следующем смысле - агенты, обученные выполнять навигацию до точки на наборе данных HM3D, достигают наивысшей производительности независимо от того, оцениваются ли они на данных HM3D, Gibson [32] или MP3D [33]. Подобное утверждение не может быть сделано про обучение на других наборах данных, где обученные агенты при переносятся на другие наборы данных теряют качество.

Затем вводятся основные понятия используемые в обучаемых навыках, которые основываются на методах обучения с подкреплением: марковский процесс принятия решения, вознаграждение, V и Q функции ценности, уравнение Беллмана. Затем описываются основные алгоритмы которые будут использоваться или модифицироваться в последующих главах, это глубокое Q-обучение (Q-Learning [23]), алгоритм актор-критик (A2C [34]) и алгоритм оптимизации ближайшей стратегии (PPO [35]).

После описывается иерархический подход в обучении с подкреплением. Иерархический подход позволяет разделить сложную задачу на множество подзадач. Наиболее известные существующие алгоритмы HRL, которые могут изучать многоуровневые иерархии, не могут эффективно изучать уровни

стратегий одновременно, особенно в непрерывном пространстве состояний и действий. Чтобы решить эту проблему, был представлен подход иерархического актора-критика (Hierarchical Actor-Critic, HAC [29]), который используется во второй главе.

Далее в главе разбирается несколько подходов решения задачи навигации. Первый рассматриваемый вариант решения задачи навигации — это использование методов обучения с подкреплением. Большим прорывом в подходах с использованием обучения с подкреплением в навигационных задачах стал метод DDPPO [21], в основе которого лежит PPO [35]. Без каких-либо модулей построения карты или планирования, DDPPO смог решить задачу навигации до точки с производительностью на уровне человека.

Второй подход, классический, предполагает декомпозицию задачи навигации на следующие подзадачи: локализация, картирования, планирование пути и следование по пути. Задачи локализации и картирования часто решаются парно, используя подход одновременной локализации и картирования (SLAM [16]).

Многие работы показали, что агенты, обученные на основе RL, уступают в избегании столкновений и управлении памятью при навигации, но превосходят в обработке неоднозначных ситуациях и лучше ориентируются в зашумленных условиях.

Третий подход, является объединением обучаемых и необучаемых методов - это разработка модулей навигации на основе обучения, которые могут быть интегрированы в единый алгоритм с классическими методами планирования. Основанные на данном подходе методы PONI и SemExp показали свое преимущество перед обучаемыми и классическими подходами в задаче навигации до целевого объекта. В предлагаемом в четвертой главе подходе SkillFusion [5] выделяются навыки агента, которые не делятся на только классические или обучаемые, а имеют обе реализации. Это позволяет агенту определять, какой тип навыка должен быть выполнен в каждый момент времени на основе его функции полезности.

Вторая глава. Навигация в реальном времени, планирование маршрута, локализация и избегание препятствий являются ключевыми задачами мобильных роботов [36].Часто эти задачи решаются методами, которые не используют машинное обучение в своей основе [37—39]. В последние годы ме-

тоды глубокого обучения и обучения с подкреплением (RL) привнесли значительные улучшения в методы навигации мобильных роботов. В данной главе рассматривается ключевая основанная на зрении задача в области мобильной робототехники - навигация внутри помещений с использованием RGB-D изображений [3] [1] [2]. Решение этой задачи предполагает анализ трех подзадач: семантическая сегментация объектов на сцене, локализация и картирования, исследование сцены и планирование маршрута.

Предлагаемое решение основано на известной парадигме sim2real, когда нейронные сети предварительно обучаются с помощью данных из симулятора, а затем полученные модели используются на реальном роботе. Для этого были собрали уникальные наборы данных на основе доступной симуляционной среды Habitat, которые позволили нам построить эффективные модели для решения проблем сегментации объектов и локализации на карте [2; 3]. Предложенный метод HISNav и собранные для него наборы данных доступны публично по ссылке [2] под лицензией MIT.

Набор данных HISNav состоит из различных траекторий движения робота, записанных в виртуальной среде Habitat. Траектории построены на 49 уникальных сценах из Matterport3D [33], которые представляют собой помещения различных типов. У каждой сцены имеется не более 5 траекторий с тремя разными видами шума в изображениях камеры и шума в действиях.

Методология. Описанный набор данных обладает рядом особенностей. Объекты в нем могут быть как небольшими, так и занимать значительную часть кадра. При этом требуется сегментировать как перемещаемые элементы интерьера (например, стул, стол, полотенце и т.д.), так и неподвижные, фоновые объекты (стена, пол, окно, лестница и т.д.). В таких условиях двухстадийные методы сегментации, которые сначала обнаруживают ограничивающие прямоугольники, а затем их сегментируют, могут работать неэффективно. Для проверки этого были рассмотрены различные архитектуры популярной модели MaskRCNN [40] с основной сетью с малым числом параметров, а также их современный аналог YOLACT++ [41]. Для сравнения качества целесообразно провести анализ получаемых метрик средней точности обнаружения объектов (mAP) [42] с различными порогами метрики коэффициента перекрытия окон (IoU).

[2] https://github.com/cds-mipt/HISNav

Для решения задачи локализации робота в пространстве по RGB-D изображениям в качестве базовых были выбрали два современных метода с открытым исходным кодом: OpenVSLAM и DXSLAM. Оба они относятся к косвенным разреженным подходам, оценивающим позу камеры (робота) по найденным ключевым точкам и их дескрипторам: в первом случае методом ORB, во втором случае нейросетевым методом HF-Net [43].

Вычисление позы текущего кадра относительно предыдущего включает в себя сопоставление ключевых точек и затем решение задачи блочного уравнивания (bundle adjustment) . Сопоставление ключевых точек происходит на основе их дескрипторов и, как правило, для ускорения этого процесса применяется мешок слов (bag of words). Кроме того, современные методы для дополнительного ускорения этапа сопоставления также используют модель движения.

При использовании модели движения метод прогнозирует перемещение агента на основе предыдущих перемещений. А именно строится матрица скорости V, а поза следующего кадра P_{cw}^{i+1} оценивается на основе позы предыдущего кадра P_{cw}^{i} по следующей формуле: $P_{cw}^{i+1} = VP_{cw}^{i}$. После оценки P_{cw}^{i+1} производится проектирование ключевых точек предыдущего кадра на текущий и ищется их сопоставление с ключевыми точками текущего кадра на основе их расстояния на плоскости изображения. Предлагаемый подход с заменой модели движения на модель управления позволяет интегрировать априорные знания о перемещении в методы SLAM и улучшает результаты визуальной локализации агента в среде.

После решения задач сегментации объектов и локализации агента на карте, необходимо решить ключевую подзадачу — навигацию к целевому объекту. Для этого был применен модульный иерархический подход, который позволяет разделить задачу навигации на планирование пути на коротком горизонте и определение подцелей на длинном горизонте. Дополнительным преимуществом предлагаемого подхода является отсутствие сложной функции вознаграждения в процессе обучения, которая обычно формируется вручную и не может полностью отразить все особенности среды.

Основная цель разрабатываемого метода — обучение стратегии Π_{k-1} k-го уровня. Где k был взят равным 2. Каждый уровень стратегии представляет из себя $\pi_i : S_i \times G_i \rightarrow A_i$. Чтобы изучить эти стратегии π_0,π_1, был исполь-

зован универсальный марковский процесс принятия решений (UMDP) U_0,U_1, где $U_k = (S,G,A,R,\gamma)$, где γ – коэффициент дисконтирования. Пространство подцелей G состоит из координат. Пространство действий первого верхнего слоя A представляет собой координаты подцели, необходимые для достижения цели задачи. Стоит отметить, что подцель первого уровня определяется относительно текущего положения агента и ограничена определенной областью $Targ$ вокруг агента, чтобы быть достижимой в любой ситуации. Пространство действий второго нижнего уровня представляет собой действия, которые агент должен выполнить для достижения подцели, определенной первым уровнем. После того, как подцель установлена первым уровнем стратегии, второму уровню стратегии необходимо определить последовательность действий N^a_{max} для достижения заданной подцели. Если эта подцель была достигнута, агент получает вознаграждение R равное 0, в противном случае -1. Первый слой имеет максимум N^t_{max} попыток для достижения конечной цели.

Нижний уровень стратегии за счет дискретного пространства действия основан на DDDQN[44], высший уровень стратегии имеет непрерывное пространство действия (координаты подцели) и основан на Twin Delayed DDPG (TD3) [45] алгоритме. Структура нейронной сети показана на Рис. 1.

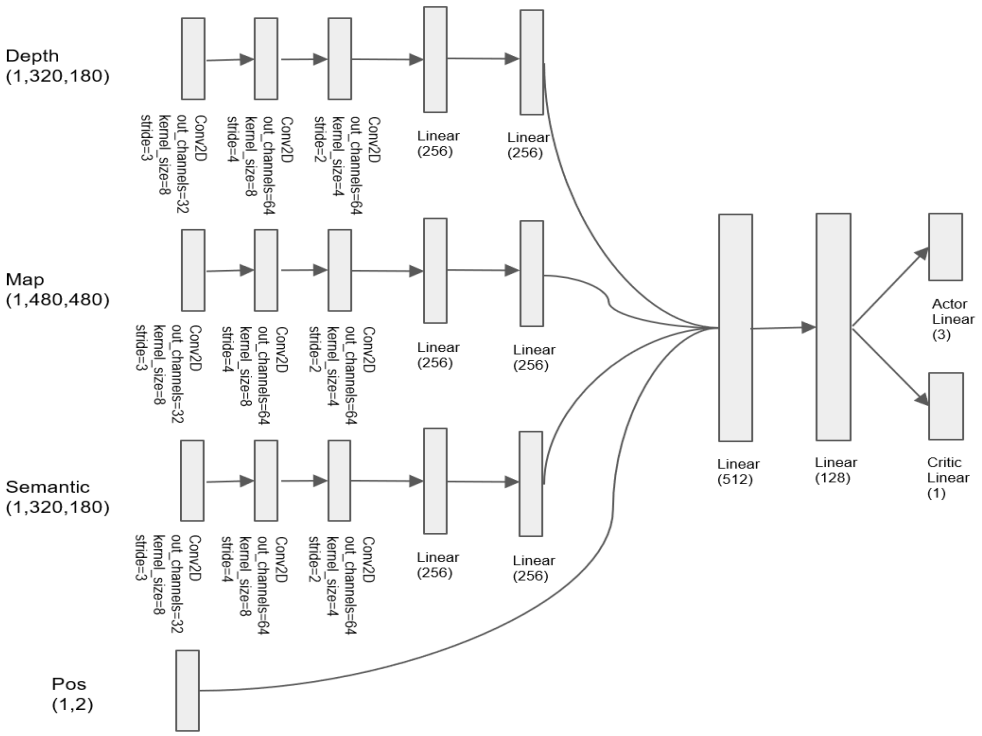

Рис. 1 — Структура нейронной сети для аппроксимации стратегии агента в методе HISNav

Чтобы обучать все уровни стратегий одновременно, была использована основная идея иерархического актора-критика (Hierarchical Actor-Critic, [29; 6]). Вместо того, чтобы оценивать действия по отношению к более низкому уровню стратегий, в данном методе оцениваются действия относительно того, куда в действительности пришел нижний уровень — оптимальная иерархия нижнего уровня. Оптимальная иерархия нижнего уровня, состоящая из оптимальных версий всех стратегий нижнего уровня, не меняется со временем. В результате состояния и вознаграждение за любое действие будут стабильными, что позволит иерархическому агенту параллельно изучать несколько уровней стратегий.

Оценка качества сегментации экземпляров по ходу обучения на валидационной выборке показана на Рис. 2.

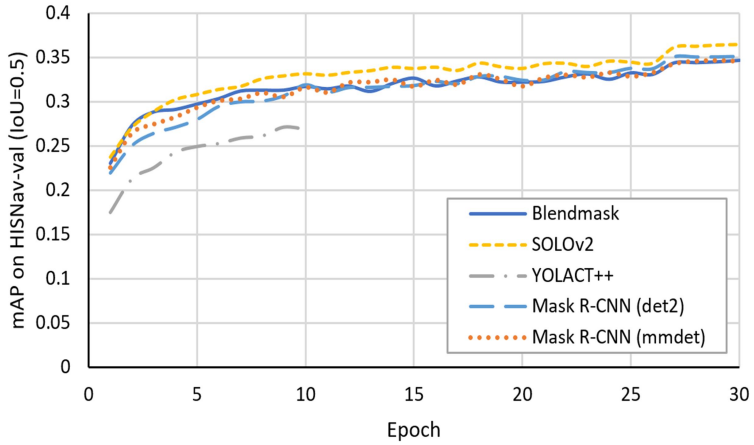

Рис. 2 — Результаты обучения моделей сегментации экземпляров.

На основе экспериментов можно сделать вывод, что наиболее высокие показатели качества у сети с архитектурой Blendmask, немного хуже результаты сегментации объектов у модели SOLOv2. Именно эти две модели могут рассматриваться как базовые для интеграции в предлагаемый метод HISNav.

Предлагаемый метод DXSLAM (названный CDXSLAM) демонстрирует лучшую производительность с точки зрения относительных показателей T_{KITTI} и R_{KITTI}.Можно сделать вывод, что интеграция данных в DXSLAM на основе нейронных сетей и в OpenVSLAM приводит к значительному увеличению относительных и абсолютных показателей качества. Визуализация одного из треков из набора данных HISNav представлена на рисунке 4.

Рисунок 3 демонстрирует эффективность обнаружения ключевых точек рассмотренными методами SLAM. Как видно на рисунке, на ключевом и

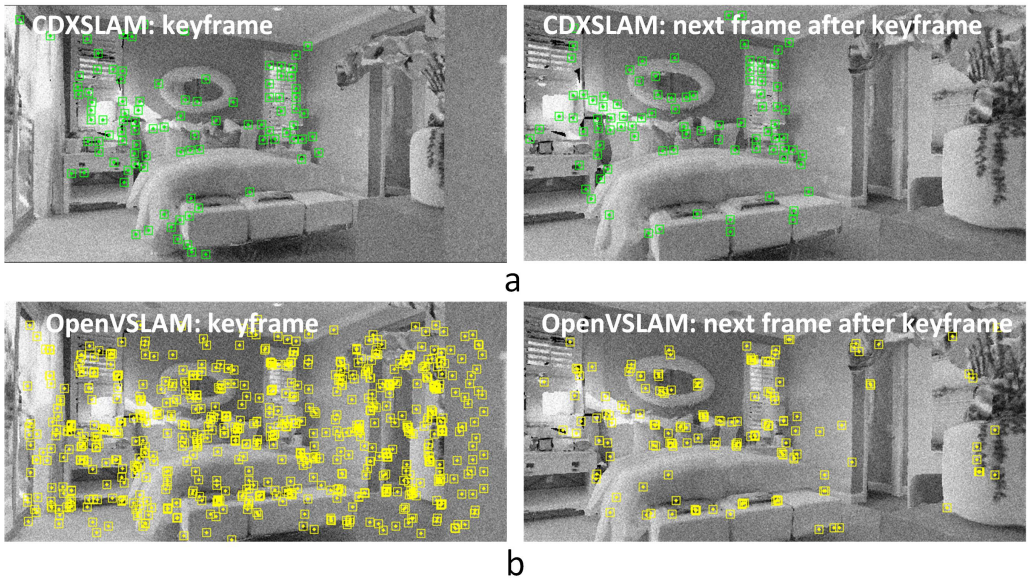

Рис. 3 — Детали обнаружения и сопоставления ключевых точек в зашумленных изображениях: а - для метода CDXSLAM, б - для метода OpenVSLAM.

последующем кадре с применением нейросетевого CDXSLAM нет шумовых точек. OpenVSLAM добавляет много шумовых точек в кадр, которые в то же время не совпадают со следующим кадром.

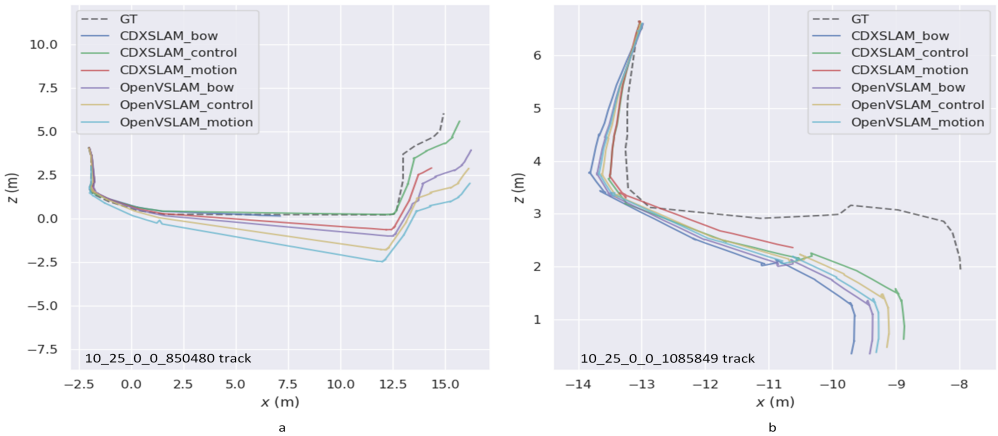

Рис. 4 — Результаты изученных методов CDXSLAM и OpenVSLAM на некоторых траекториях набора данных HISNav. Интеграция в модель движения данных управления повышает качество обоих методов.

Методы глубокого обучения с подкреплением часто сталкиваются с проблемой недостатка вознаграждения за исследование, что замедляет процесс обучения на больших сценах. Один из подходов, решающих эту проблему, — это метод случайной дистилляции сети (Random Network Distillation,

[46]), который был выбран для сравнения с HISNav методом. В общем случае, задача навигации к целевому объекту сводится к двум подзадачам: навигации к точке и исследованию сцены. Предлагаемая архитектура успешно справляется с задачей навигации к точке, а для решения задачи исследования был использован модуль глобальной стратегии SemExp [47]. Для ускорения процесса обучения, во всех приведенных ниже экспериментах была использована одна сцена с несколькими эпизодами, имеющими различные начальные и конечные точки.

В следующем эксперименте, приводится сравнение PPO, RND и HISNav методов в задаче навигации до точки с наличием Гауссова шума сенсоров ($mean = 0$, $\sigma = 1$, $intensity = 0.05$) и Гауссова шума в действиях агента ($mean = 0$, $\sigma = 1$, $intensity = 0{,}2$).

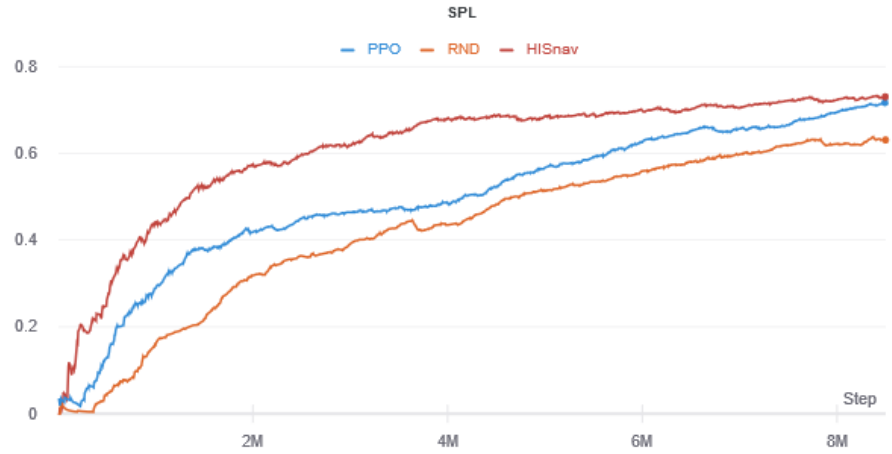

Рис. 5 — PPO vs RND vs HISNav в задаче навигации к точке.

Как показано на рисунке 5, метод RND демонстрирует результаты, которые немного уступают PPO. Внутреннее вознаграждение RND не обеспечивает прироста в начале процесса обучения, и этот прирост снижается до минимума к концу. Предлагаемый метод в конечном итоге сходится к значению SPL, равному 0,7, для PPO и RND, однако он достигает порога в SPL 0,6 в два раза быстрее, на 3M, а не на 6M.

В **третьей** главе была изучена задача визуальной навигации в помещении к объектам, которые определяются их семантической категорией [48]. Недавние работы показали значительные достижения в end-to-end подходах к обучению с подкреплением и модульных системах. Однако оба подхода нуждаются в большом шаге вперед, чтобы быть устойчивыми и практиче-

ски применимыми. Дальнейшее увеличение показателей метрики невозможно, поскольку нерешенные эпизоды представляют собой обширные области, исследование которых без предварительной информации о их семантической структуре представляется затруднительным. Предполагается, что такая информация может быть легко аннотирована человеком и она должна оставаться неизменной в областях с часто перемещающимися объектами. Следовательно, агент должен обладать иерархической структурой для эффективного использования пространственного понимания. Чтобы решить проблему недостаточного исследования сцен и сделать исследование более семантически значимым, предлагается расширить стандартную постановку задачи и дать агенту легкодоступные ориентиры в виде расположения комнат и их тип. Наличие ориентиров позволяет агенту построить иерархическую структуру стратегии и добиться успеха в 63% на валидационных сценах в фотореалистичном симуляторе Habitat. В используемой иерархии, низкий уровень состоит из отдельно обучаемых навыков, а высокий уровень решает, какой навык необходим в данный момент. Также в данной главе показывается возможность переноса обученного алгоритма на реального робота. После небольшого дообучения на реконструированной реальной сцене, робот показывает до 79% SPL при решении задачи навигации к произвольному объекту.

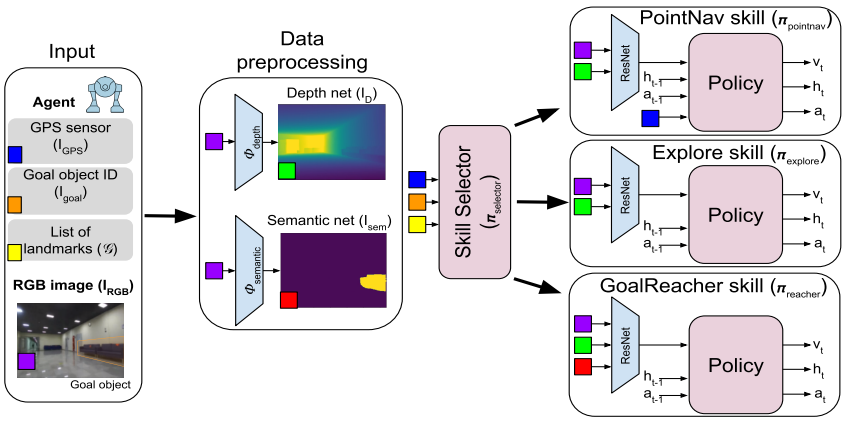

Рис. 6 — Схема метода иерархической стратегии с ориентирами (HLPO). Предлагаемый нами подход состоит из трех основных блоков: предварительная обработка данных, селектор навыков и стратегии навыков. Разноцветные квадраты внизу элементов означают, какие данные отдают модули на выходе (внизу слева) и принимают в качестве входа (внизу справа).

В данной главе предлагается основанная на ориентирах модульная структура (рис. 6) для перехода к цели (I_{goal}) – Иерархическая оптимизация ориентиров (HLPO). Предлагаемый подход состоит из трех основных модулей: модуль выбора навыков $\pi_{selector}$, модуль предварительной обработки данных $\Phi_{semantic}$ и Φ_{depth} и стратегии навыков $\{\pi_{explore}, \pi_{reacher}, \pi_{pointnav}\}$.

Модуль выбора навыков анализирует все тренировочные сцены, связывая типы объектов с типами комнат, и составляет статистику. Для сбора данной статистики, информация о том, какие объекты принадлежат к какому типу комнаты была собрана со всех сцен набора данных Matterport. Основываясь на этой статистике, текущем типе цели объекта и расстояниях до комнат в текущий момент, модуль селектора умений ранжирует комнаты (как вероятность нахождения объекта в комнате/расстояние до комнаты) в порядке необходимости посещения для поиска целевого объекта.

Модуль предварительной обработки данных представляет собой две нейронные сети, выполняющие семантическую сегментацию и реконструкцию глубины.

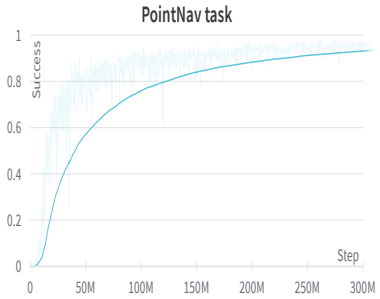

Рис. 7 — Доля успешных выполненных эпизодов при выполнении навыка PointNav.

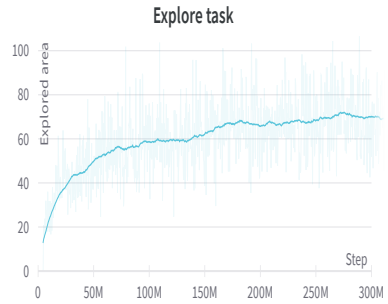

Рис. 8 — Исследованная область (m^2) при выполнении навыка Exploration.

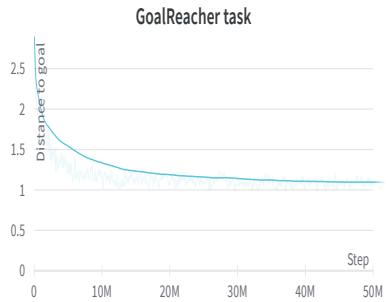

Рис. 9 — Расстояние до целевого объекта (m) при выполнении навыка GoalReacher.

Модуль выбора навыков принимает в качестве входных данных наблюдение агента и выводит действие, направленное на получение текущего необходимого навыка. Для сформулированной задачи были определены три навыка: навигация до заданных координат (PointNav), исследование сцены (Exploration) и достижение цели (GoalReacher). В проведенных экспериментах, навык PointNav выполнялся в 50% случаев во время эпизода, навык Exploration — в 22%, а навык GoalReacher — в 28%.

Навык PointNav обучался для достижения ориентира (комнаты представляющий наибольший интерес исходя из типа цели) (Рис. 7). Вознаграждение давалось пропорционально сокращенному расстоянию до цели. Стратегия принимает изображение RGBD и координаты относительно ориентира в качестве входных данных.

Навык GoalReacher был обучен навигироваться к объекту на заданное расстояние, когда его видит семантический сенсор (Рис. 8). Чтобы избежать переобучения, агент во время инициализации эпизода появлялся в случайном месте, а затем ему задавалась цель достичь самого дальнего видимого объекта. Вознаграждение также пропорционально сокращению расстояния до ближайшего объекта целевого типа.

Навык Exploration больше других определяет успех задачи, особенно когда у агента нет ориентиров. В данной работе, для этого навыка агента была обучена стратегия на основе обучения с подкреплением, которая эффективно исследует близлежащую область (Рис. 9), поэтому она идеально подходит для полного исследования комнаты, но имеет более низкий процент охвата в больших сценах. В качестве вознаграждения агент получает +1 каждый раз, когда входит в новую зону размером в один квадратный метр.

Процесс управления навыками можно рассматривать как стратегию и реализовать как модуль глобальной стратегии. Как правило, методы HRL могут обучаться двухуровневой стратегии Π_1. Каждый уровень стратегии изучает $\pi_i : S_i,G_i \rightarrow A_i$, где G_i — множество возможных подцелей. Для изучения этих стратегий π_i используется множество MDP U_0,U_1, в которых $U_k = (S,G,A,T,R,\gamma)$. Для ObjectGoal задачи последовательность навыков может быть сформулирована в явном виде. Агент перемещается к первому ориентиру (координате интересующей комнаты, заданной глобальной стратегией) с помощью навыка PointNav. Затем агент исследует комнату, пока не покинет ее с помощью навыка Exploration. Если модель семантической сегментации видит целевой тип объекта в интересующей комнате, активируется навык GoalReacher и осуществляется переход к этому объекту.

Для сравнения метода HLPO с существующими современными подходами были выбраны пять базовых решений, которые не используют инфор-

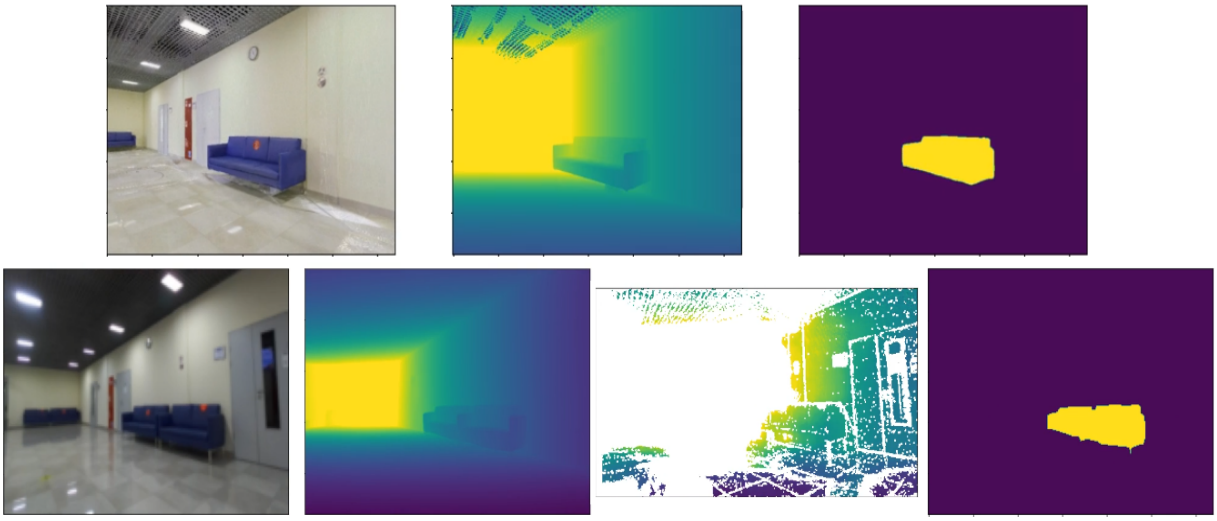

Рис. 10 — Верхний ряд — данные, подаваемые на вход агенту в симуляторе. Изображение по центру верхнего ряда — это реальная глубина, которая была ограничена пятью метрами. Нижний ряд — то что видит агент в реальности. Второе изображение нижнего ряда — это глубина, полученная модулем реконструкции глубины. На третьем изображении нижнего ряда показано сравнение качества глубины нейронной сети с глубиной камеры ZED. Оба правых изображения представляют собой семантическую маску класса дивана, полученную модулем семантической реконструкции.

мацию об ориентирах. Однако на этапе обучения у них была поставлена цель изучить семантическую структуру сцен.

В результате применения подхода, основанного только на навыках Explore и GoalReacher, были получены результаты, сопоставимые с SOTA методами. Использование полного метода HLPO почти удвоило количество успешных эпизодов при валидации, что указывает на правильность предлагаемого выбора информации об ориентирах в качестве дополнительной необходимой информации о сцене, и на то, что end-to-end подходы не могут самостоятельно изучить ее.

Для доказательства того, что шумы датчиков и действий не существенно ухудшают результаты навигации, шумы были добавлены и на этапе обучения стратегий, благодаря чему агент научился не обращать на них большого внимания. Проведенные эксперименты (строка HLPO (Noise) в таблице 1) подтвердили это. Эксперимент с добавлением полной карты (строка HLPO (Map) в таблице 1) показал, что общая эффективность метода может быть повышена, если у агента есть доступ к полной карте препятствий. Однако

Method	GT semantic		Learned semantic	
	Success	SoftSPL	Success	SoftSPL
E2E RL	0.18	0.35	0.11	0.24
SemExp	0.24	0.26	0.11	0.17
Planning	0.31	0.26	0.15	0.18
Auxiliary RL	0.51	0.34	0.19	0.19
ExploreTillSeen	0.46	0.33	0.20	0.21
HLPO	**0.86**	**0.52**	**0.46**	**0.30**
HLPO (Noise)	0.85	0.46	0.45	0.28
HLPO (Map)	0.90	0.54	0.51	0.42

Таблица 1 — Сравнение различных агентов на тренировочных эпизодах.

выигрыш не так заметен по сравнению со сложностью сбора полной карты для каждой сцены.

Существует несколько способов обучения робота задаче навигации: от обучения в реальных сценариях до обучения в полностью смоделированных средах с последующим переносом полученной стратегии на реального робота. Первый слишком неэффективен, так как требует много ресурсов и может принести много вреда, пока стратегия не оптимальна. С последним удобно работать, и можно получить высокие результаты в симуляционных средах, но общая задача — сделать так, что бы агент мог ориентироваться в реальных сценах. В этом случае процесс перехода от смоделированной среды к реальному роботу может быть даже более сложным, чем само обучение, так как в симуляторе агент не сталкивается с многими недостатками датчиков реального мира.

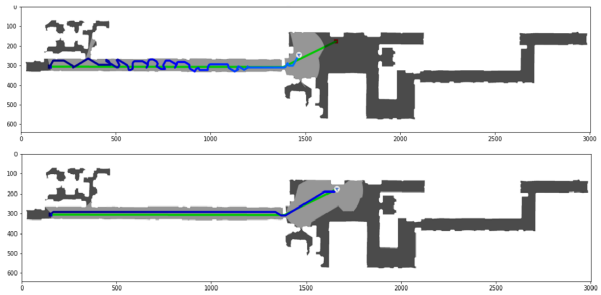

Рис. 11 — Сравнение агента HLPO до и после адаптации.

Агент был обучен на максимально возможном количестве сцен, полученных из набора данных Matterport. Однако разнообразие сцен во всех наборах данных настолько велико, что 60 сцен недостаточно для учета всех

Рис. 12 — На нижнем изображении показано, как выглядит облако точек до нанесения текстур. Верхнее изображение это облако точек после нанесения текстур в программе RealityCapture.

характеристик любой сцены. Это и произошло в случае проводимых экспериментов в лабораторной зоне. Длинные одинаковые коридоры практически отсутствовали в обучающем наборе данных, и поведение агента внутри них значительно отклонялось от оптимального пути (Рис. 11). Поскольку дополнить набор данных большим количеством подобных сцен затруднительно, было решено перенастроить агента на реконструированную сцену перед проведением реальных тестов. Данный подход является подходящим для внутренней навигации в подобных случаях, поскольку такая калибровка требуется один раз для каждого типа сцены, и дальнейший пользователь системы может загрузить соответствующие веса для своего типа сцены.

Создатели исходного набора данных mp3d [33] использовали камеру Matterport Pro2 (134 мегапикселя без лидара) и фирменное программное обеспечение Matterport, которое позволяет загружать фотографии для автоматической обработки в окончательный файл .obj. Этот файл затем можно использовать в симуляторе Habitat. Этот подход работает хорошо, если сцена не содержит мелких деталей. Однако в самом наборе данных mp3d было много пробелов и несоответствий текстур. В данной работе было стремление к более высокому качеству сцены, особенно к более точной реконструкции глубины, чтобы агент мог предсказуемо ориентироваться в узком пространстве комнаты. Предлагаемое решение заключается в использовании лазерного сканера Leica RTC360 3D. Дополнительным преимуществом является возможность предварительного редактирования полученных кадров, удаления проходящих людей и проверки качества сборки всей сцены. Для нанесения тек-

стур на конечное облако точек была использована программа RealityCapture [3] (Рис. 12).

Рис. 13 — Вероятность успеха во время адаптации сцены.

Кроме того, реконструированная сцена позволила откалибровать нейронную сеть по восстановлению глубины (Рис. 10) и установить правильные ограничения. Таким образом, глубина, полученная нейронной сетью, получилась очень близка к глубине, полученной из симулятора. Затем на полученной сцене агент был дообучен в течении одного миллиона шагов (Рис. 13)). Благодаря этому были учтены все особенности сцены.

После адаптации метода HLPO под особенности валидационной сцены, вероятность успеха выполнения эпизода в реконструированной сцене увеличилась с 0,6 до 0,9. А на реальном тесте с роботом Clearpath Husky был получен SPL равный 0,79, что говорит о том, что предлагаемый метод HLPO одинаково хорошо работает как на симуляторе, так и в реальных тестах.

В **четвертой** главе предлагается гибридная схема, состоящая из модулей, основанных на необучаемых и обучаемых методах, а также переключателя между ними – SkillFusion. Первые более точны, в то время как вторые более устойчивы к шумам датчиков. Чтобы уменьшить разрыв между симуляцией и реальностью, который часто возникает при использовании обучаемых методов, предлагается обучать их таким образом, чтобы они были менее зависимы от окружающей среды. В результате, метод показал лучшие результаты как в симуляторе Habitat, так и во время проверки на реальном роботе.

ObjectGoal является сложной задачей, которая требует различных видов поведений от агента. Эти виды поведения можно рассматривать как навыки агента. Для ее решения, были выделены два основных навыка, которые напрямую решают задачу ObjectGoal: навык исследования (Exploration)

[3] https://www.capturingreality.com/

и навык достижения цели (GoalReacher), а также два дополнительных: навык достижения заданной точки (PointNav) и навык бегства (Flee). Каждый из основных навыков был реализован как классическим, так и обучаемым методом. Дополнительные навыки используются для обобщающего эффекта обучаемой стратегии навыка Exploration.

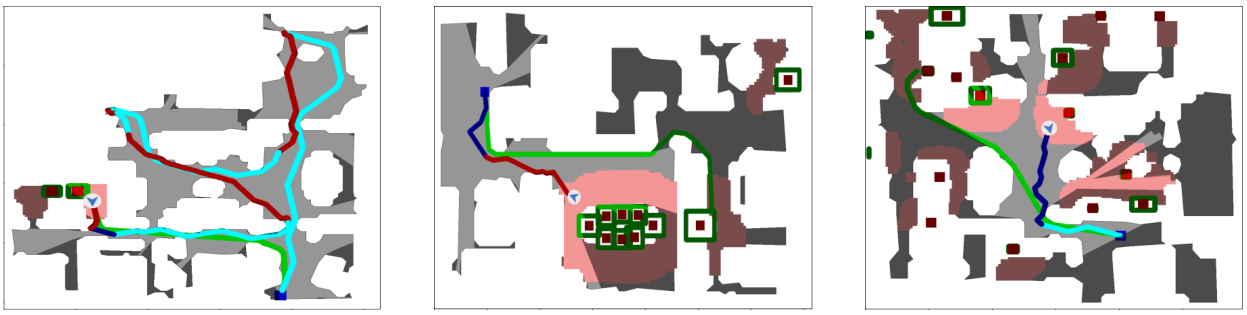

Рис. 14 — Примеры работы SkillFusion в симуляторе. Красная линия - траектория, полученная классическим методом, светло-синяя линия - обучаемым навыком Exploration, а темно-синяя - обучаемым навыком GoalReacher.

Реализация классических навыков Exploration и GoalReacher состоит из следующих частей: постановка промежуточной цели, локализация, картирования, планирование пути и управление движением.

Модуль постановки промежуточной цели. Модуль постановки промежуточной цели принимает текущую оценку положения робота и карту, построенную с помощью SLAM, и выбирает цель, куда должен направиться робот. Если целевой объект отображен на карте SLAM, то выбирается ближайшая к нему точка.

В качестве модуля постановки промежуточной цели, была использована реализация подхода к исследованию среды на основе границ, которая была предложена в работе [49]. Алгоритм ищет границы между свободным и неизвестным пространством на 2D карте, построенной с помощью SLAM. Для поиска этих границ используется поиск в ширину (BFS).

Всем границам присваиваются функции стоимости, а центроид границы с наименьшей функцией стоимости отмечается как цель для робота. Функция стоимости границы учитывает расстояние между агентом и центроидом границы, размер границы и угол поворота агента перед тем, как он начнет движение к границе.

Модуль локализации. В симуляции используются данные о точном положении робота, поэтому внешний модуль локализации не требуется. На реальном роботе, для максимальной точности локализации и избежания накопления ошибок используется алгоритм лидарной одометрии LOL-odom [50] с дополнительной коррекцией положения через предварительно построенную 3D карту (эта карта используется только для коррекции положения).

Модуль картирования. Модуль картирования строит карту занятости сетки для навигации на этапе исследования сцены и семантическую карту для навигации к целевому объекту. В симуляции, где доступны данные о точном положении, для построения карты занятости используется обратную проекцию карты глубины. Для построения семантической карты, используется обратная проекция предсказанной семантической маски. Все объекты высотой менее 0,2 м отображаются как ячейки пола, а все объекты выше 0,2 м отмечаются как препятствия. На реальном роботе используется алгоритм SLAM RTAB-MAP [51] с лидаром в качестве источника данных. Семантическая информация добавляется на карту RTAB-MAP из семантических масок, предсказанных с бортовой RGBD-камеры.

Модуль планирования. Для планирования пути был использован алгоритм Theta* [18]. Это алгоритм поиска пути на графе, который поддерживает перемещение агента под любым углом. По умолчанию Theta* планирует путь для агента нулевого размера, поэтому он был модифицирован для учета размера робота. В предлагаемой модификации, робот моделируется как диск радиуса r. На практике размер диска устанавливается больше, чем фактический размер робота, чтобы был запас по безопасности. Если путь не найден, то безопасный радиус уменьшается до $0.8r$ и повторяется планирование пути. Для проводимых экспериментов, $r = 0.15$ для симуляции и $r = 0.6$ для робота.

Модуль контроллера. В симуляторе выбранное действие выполнялось без дополнительных обработок в дискретном виде, а на реальном роботе была использована реализация метода следования траектории [4], которая получает путь и положение робота, и выдает команды скорости для робота. Это следование траектории похоже на подход к управлению в симуляции, но оно

[4]https://github.com/andrey1908/strl_robotics/tree/main/control

адаптировано для непрерывного движения и имеет некоторые эвристики для компенсации ошибок одометрии.

Для навыка GoalReacher были использованы те же модули локализации, планирования, картирования и контроллера, которые описаны выше. Для достижения нанесенной на карту цели был использован подход на основе границ, но вместо границ между свободным и неисследованным пространством карты используются границы целевых объектов.

Так же, как и классические, обучаемые навыки были разделены на две части: стратегия исследования среды (Exploration) и стратегия достижения цели (GoalReacher). Для обучения этих навыков была использована децентрализованная распределенная оптимизация ближайшей стратегии (DD-PPO), так как она показывает одни из лучших результатов в аналогичных задачах визуальной навигации [21]. Архитектура для обучения предложенных обучаемых навыков показана на рис. 15. Ключевым фактором для этого результата было огромное количество шагов обучения. Это требует быстродействующего симулятора, который также должен быть фотореалистичным, для возможности переноса полученной стратегии в реальный мир. В этой связи в данной работе также был использован самый быстрый из доступных фотореалистичных симуляторов, BPS, [28] с самым большим набором данных из 1 000 сцен, HM3D [31]. Для обучения обучаемого навыка Exploration использовалась обучающая часть набора данных HM3D (800 сцен) и 145 сцен для обучаемого навыка GoalReacher, так как в HM3D есть только это количество сцен с доступной семантикой.

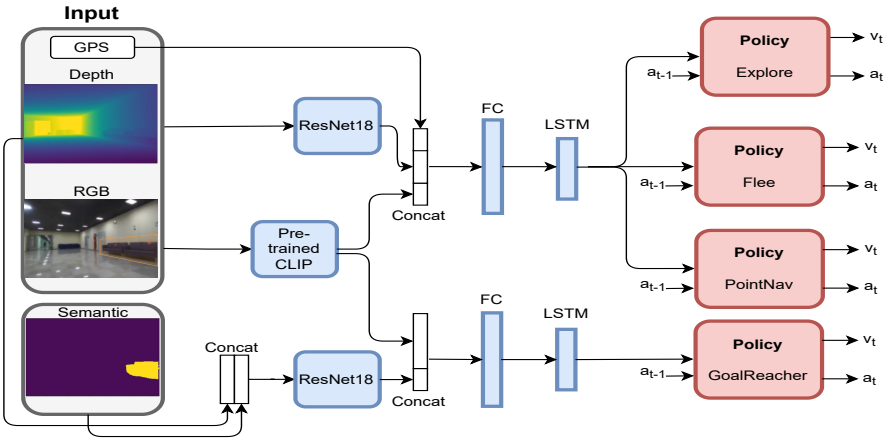

Рис. 15 — Предложенная архитектура нейронной сети метода SkillFusion для синтеза задач навигации.

Исследование сцены является сложной задачей, которая требует от стратегии агента точности, способности запоминать прошлый опыт и планировать будущее. В реальных и фотореалистичных средах, эта задача усложняется значительным разнообразием размеров, планировки, обстановки, цвета и других параметров внутренних сред. Чтобы обобщить эти особенности, агенту необходим мощный кодировщик, чтобы извлечь всю необходимую информацию из шумного RGB-датчика. Решение этой проблемы состояло из двух частей: получение визуально устойчивого кодировщика изображений и разработка динамически устойчивого RNN-кодировщика.

В качестве кодировщика изображений была использована предварительно обученная модель CLIP [52] и заморозили ее во время фазы обучения. Этот подход использовался в предыдущих работах [53] [54], и проведенные в данной работе эксперименты также демонстрируют способность модели навигации исключительно на основе RGB-датчика с замороженным кодировщиком CLIP.

Чтобы решить проблему динамически устойчивого RNN-кодировщика, все навигационные навыки были объедены и обучены с помощью одной нейронной сети (Рис. 15) с общими визуальными кодировщиками и слоями RNN и несколькими головами [55]. Навык Exploration должен быть осведомлен о пространственной структуре помещения, чтобы избежать исследования уже исследованных областей, в то время как навык Flee фокусируется на измерениях расстояний и ориентаций, навык PointNav фокусируется на поиске кратчайшего возможного пути к точке. Функция вознаграждения каждой задачи мотивирует эти поведения.

Навык GoalReacher. Важным аспектом навыка GoalReacher является способность различать целевые и нецелевые объекты. Чтобы передавать модели информацию об объекте, используется бинарная маска сегментации объекта, а не его идентификатор. Этот подход позволил обучать модель семантической сегментации независимо от стратегии агента. Недостатком этого является то, что нужно обучать стратегию с истинным датчиком семантики в симуляторе, чтобы иметь возможность давать правильные вознаграждения.

Во время проверочных эпизодов модель семантической сегментации может производить шум и ложноположительные объекты, которые появляются с некоторой периодичностью. Навык GoalReacher обучается с большим

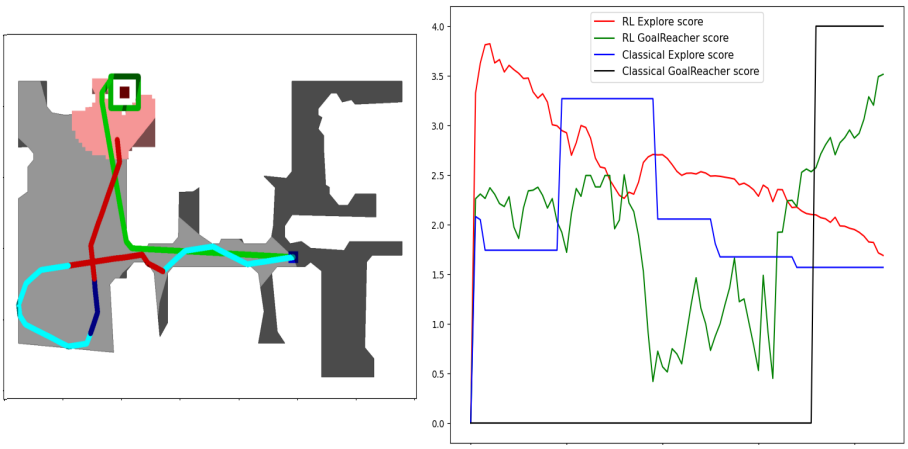

Рис. 16 — Пример управления навыками во время эпизода. Синий цвет траектории обозначает выполнение обучаемого навыка Exploration, темно-синий обозначает обучаемый навык GoalReacher, а красный цвет обозначает классические навыки.

положительным вознаграждением за успешное завершение эпизода и малым отрицательным вознаграждением за каждый выполненный шаг. В результате шумные объекты могут заставить агента завершить эпизод в месте, где появляется шумный объект, в надежде, что это настоящий, что приведет к большому отрицательному вознаграждению, если это не так. Или же агент может попытаться найти другой более далекий объект и накопить большую сумму малых отрицательных вознаграждений, выполнив слишком много действий.

Чтобы позволить агенту различать шумные и реальные объекты и избежать попадания в неблагоприятные ситуации принятия решений, предлагается использовать новое действие “not sure”. Когда агент не уверен в достоверности маски сегментации, он может выполнить это действие, чтобы вернуться к навыку Exploration, а не преследовать воспринимаемый шумный объект.

Во время фазы обучения навыка GoalReacher, эпизоды с ложной семантической сегментацией объекта выбираются с вероятностью 0,2, остальные эпизоды содержат истинную сегментацию. В шумных эпизодах агент может получить небольшое отрицательное вознаграждение, если он выбирает действие “not sure”, когда завершает эпизод, и большое отрицательное вознаграждение, если он выполняет действие “stop” в любом другом месте. Цель агента - быстро распознать, является ли семантическая сегментация ложной,

и завершить эпизод с действием “not sure”, или преследовать цель, убедиться в ее достоверности и выполнить действие “stop” рядом с ее положением.

Чтобы объединить преимущества классических и обучаемых подходов, был предложен метод SkillFusion, так как, будучи отдельными, эти подходы имеют свои ограничения.

Классические методы исследования на основе границ на 2D карте надежны и устойчивы к изменениям сцены и камеры; они требуют только точных источников одометрии и карты. Однако, обучаемые подходы могут быть более эффективными при решении конкретных сложных задач, так как они могут быть обучены напрямую решать эти задачи, например, используя методы обучения с подкреплением. В задаче навигации к целевым объектам, подходы на основе RL показывают большую эффективность в сценах, где цель можно найти за короткое время, но на больших расстояниях рекуррентные нейронные сети начинают забывать информацию с первых шагов, в следствии чего агент начинает посещать уже исследованные области. Классические методы, с другой стороны, хранят всю информацию в виде 2D карты и могут исследовать новые места без повторений в течение всего времени работы. Кроме того, подходы обучения с подкреплением хороши в поиске цели и прицеливании на нее, но им трудно остановиться вблизи объекта точно в заданном радиусе.

Чтобы эффективно использовать преимущества каждого подхода, подход SkillFusion должен знать о тех преимуществах, которые каждый навык может предоставить в любой момент времени. В стратегии агента уже есть функции стоимости для каждого из навыков: функция критика для обучаемых навыков и функция стоимости границ и пути для классических навыков (Рис. 16). Данные функции нормируются до одного порядка на основе собранного опыта, и на каждом шаге агент выполняет навык с максимальным значением функции полезности. Так как целью задачи ObjectNav является выполнение действия остановки, когда агент находится рядом с целью, классическому навыку GoalReacher присваивается высокое постоянное значение, когда цель появляется на карте после всех фильтров, так как этот навык наиболее полезен для выполнения задачи, чем любые исследования сцены.

В результате, агент начинает исследование сцены с помощью обучаемых навыков для эффективного покрытия больших областей вокруг стар-

товой позиции. Спустя некоторое время после старта, обучаемые навыки забывают информацию в своих рекуррентных слоях и начинают исследовать уже исследованные области. Когда это происходит, оценка обучаемого навыка Exploration становится меньше оценки полезности классического навыка Exploration, и агент переключается на классический навык. В этот момент классический навык Exploration на основе текущей карты направляет агента к границе с наивысшей стоимостью, и после этого оценка обучаемого навыка Exploration становится больше классической, и стратегия возвращается к обучаемому навыку (Рис. 14 слева). Следует отметить, что рекуррентные слои обучаемых навыков обновляются на каждом шаге, даже когда агент действует на основе классических навыков.

Если целевой объект становится видимым для семантической сегментации, оценка обучаемого навыка GoalReacher резко повышается и берет на себя контроль агентом, пока целевой объект не будет нанесен на семантическую карту. При выполнении обучаемого навыка GoalReacher, агент имеет два варианта: завершить эпизод самостоятельно, выполнив действие “стоп” (Рис. 14 справа), или вернуть контроль навыкам Exploration, выполнив действие “не уверен”, если он решит, что целевой объект является шумом. Когда целевой объект уже находится на карте сегментации, оценка классического навыка GoalReacher устанавливается на постоянно высокое значение, агент переключается на этот навык и завершает эпизод по достижению цели (Рис. 14).

У обучаемых стратегий кодировщик RGB-кодировщик составляет значительную часть параметров нейронной сети, что замедляет фазу обучения. Кроме того, количество сцен в наборе данных ограничено, но от агента требуется, чтобы он мог навигировать за их пределами. Поэтому, обобщающая способность RGB-кодировщика в новых сценах является сложной проблемой. Решением этой проблемы может быть использование уже предварительно обученного RGB-кодировщика, который будет заморожен во время фазы обучения. В качестве такого кодировщика была выбрана модель CLIP [52]. Чтобы продемонстрировать, что сокращение обучаемых параметров все равно позволяет агенту выполнять задачи навигации, производительность задачи GoalReacher была сравнена с замороженным CLIP по сравнению с обучаемой моделью ResNet 17. Как показал данный эксперимент, производитель-

ность агента даже улучшилась. Для анализа влияния датчиков глубины на производительность и демонстрации того, что нейронная сеть опирается на встраивания CLIP не только для распознавания объектов, но и для решения задач навигации, навык GoalReacher был обучен исключительно на входных данных RGB. Предлагаемая архитектура с этим ограничением все равно показала высокий результат.

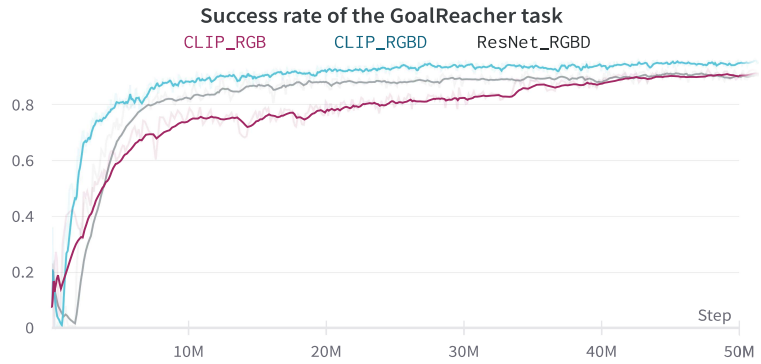

Рис. 17 — Сравнение обучения навыка GoalReacher с использованием необучаемого кодировщика изображений CLIP и обучаемого кодировщика ResNet.

Чтобы продемонстрировать необходимость слияния классических и обучаемых подходов, были протестированы различные комбинации классических и обучаемых навыков. Результаты сравнения показывают, что только обучаемый подход лучше, чем только классический подход по метрике SPL в симуляторе, но имеет сопоставимую вероятность успеха (см. Таблицу 2). Это происходит потому, что классические методы полагаются только на карту, и являются “слепыми” за пределами ограничений диапазона глубины и тратят много времени на исследование углов или тупиков, в то время как обучаемые методы могут видеть их, извлекая информацию из RGB камеры. Сопоставимая вероятность успеха обусловлена тем, что для классических методов проще решить, когда агенту нужно завершить эпизод, так как они могут рассчитать расстояние до цели с помощью метрической карты.

Как показано в таблице 3, производительность модели SkillFusion чувствительна к качеству семантической сегментации. С истинным значением (GT) семантики результаты почти вдвое превосходят результаты SegFormer [12], но с реализацией фильтрации семантической карты для классических методов и добавлением действия “не уверен” в обучаемые методы, этот разрыв был по большей части компенсирован.

Таблица 2 — Абляция различных доступных наборов навыков агента во время выполнения эпизода.

Skill		**Metrics**		
Explore	GoalReacher	Success	SPL	SoftSPL
Classical	Classical	0.410	0.182	0.263
RL	RL	0.403	0.224	0.321
RL+Classical	Classical	0.511	0.299	0.309
RL+Classical	RL+Classical	0.547	0.316	0.365

Таблица 3 — Абляции модуля семантической сегментации.

Method	**Success**	**SPL**	**SoftSPL**
SkillFusion (with no filtering)	0.324	0.207	0.327
SkillFusion (with filtering)	**0.547**	**0.316**	**0.365**
SkillFusion (ground truth semantic)	0.647	0.363	0.384

Кроме экспериментов с симуляцией, был проведен ряд тестов на роботе Clearpath Husky. От робота требовалось найти различные объекты (например, стул или диван) в неизвестной агенту среде (здание университета). Он выполнял задачи без какой-либо дополнительной настройки нейронных сетей.

Траектории для всех методов показаны на рис. 18, а значения метрик показаны в таблице 4. Были измерены пять метрик: коэффициент успешности, SPL, SoftSPL, длина пройденного роботом пути и время в пути. С только RL подходом робот дважды не смог достичь целевого объекта. В тесте с синим стулом робот достиг дивана вместо стула. Это произошло из-за ошибок семантического сегментатора и подхода к достижению цели без фильтрации семантических данных. В тесте с красным стулом робот проигнорировал узкий проход со стулом, потому что RL был обучен с функцией вознаграждения, пропорциональной исследованной области. В результате функция вознаграждения за вход в узкий проход была слишком низкой.

Таблица 4 — Результаты отдельных тестов классических и обучаемых подходов на реальном роботе в сравнении с методом SkillFusion.

Method	Success	SPL
RL	0.0	0.00
Classic	1.0	0.40
SkillFusion	**1.0**	**0.61**

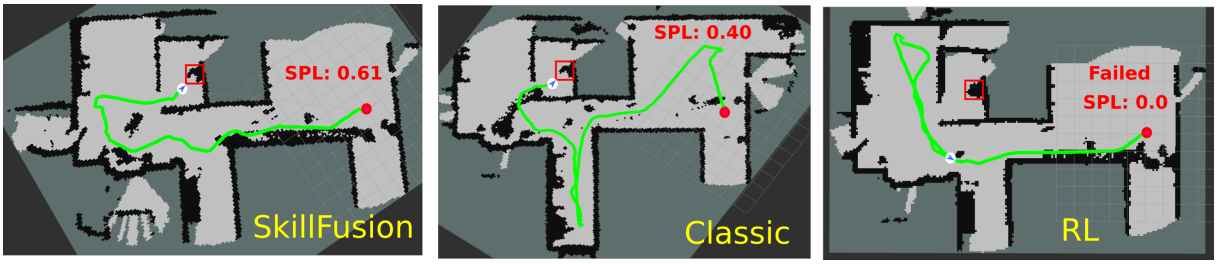

Рис. 18 — Траектории робота при использовании разных методов: SkillFusion (слева), классический метод (посередине) и обучаемый метод (справа). Красный прямоугольник обозначает целевой объект, красный круг обозначает начальную точку, а белый круг с синей стрелкой обозначает точку остановки робота и направление.

В **заключении** приведены основные результаты работы:

– Для задачи навигации до точки был предложен иерархический метод с автоматическим выбором подцелей и интеграцией алгоритмов картирования и семантической сегментации. Работоспособность данного метода проверена как в более простой 2D постановке, так и в фотореалистичном симуляторе в условиях шума данных в датчиках и актуаторах. Проведено множество численных экспериментов, показывающих производительность данного метода в сравнении с методами без выделения иерархий по скорости обучения.

– Был предложен новый подход к навигационной задаче поиска целевых объектов. Как показали проведенные численные эксперименты, при стандартной постановке задачи, существующие методы ограничены тем, что в семантически сложных сценах, исследование без априорной информации о сцене занимает в два раза больше времени по сравнению с использованием полной карты местности. Однако построение точной карты местности для каждой сцены с помощью визуальной информации на мобильном роботе не представляется возможным. Для решения этой проблемы были предложены ориентиры в виде списка координат комнат и их типа. С обновленной формулировкой задачи была создана новая иерархическая архитектура, в которой используются навыки, которые можно комбинировать и повторно использовать в различных навигационных задачах без изменений. Был разработан процесс переноса обучаемых навыков на реального робота через реконструкцию сцены в симуляци-

онную среду с фотореалистичным качеством, и дообучением в ней агента перед переносом стратегии на реального робота.

– Классические и обучаемые подходы к навигации роботов были реализованы в виде навыков агента и был предложен модуль выбора навыков, который на основе внутренней модели оценки полезности каждого навыка, делает выбор между ними во время выполнения задачи. Предлагаемый метод SkillFusion опирается на фундаментальные различия между классическими и обучаемыми методами. Проведенные численные эксперименты показали прирост доли успешных траекторий на 30% при использовании предлагаемого подхода по сравнению с классическим и обучаемым подходами. Добавление обучаемой стратегии к классической при выполнении навыка исследования сцены показало прирост на 20% за счет избегания застревания агента в узких местах, где агент не может построить точную карту препятствий. При выполнении навыка следования до цели, обучаемая стратегия показала прирост 10% итоговой метрики качества за счет способности дальнего обнаружения объектов, которые не могут быть нанесены на карту с достаточной точностью.

Публикации автора по теме диссертации

1. Learning embodied agents with policy gradients to navigate in realistic environments / A. Staroverov [и др.] // Advances in Neural Computation, Machine Learning, and Cognitive Research IV: Selected Papers from the XXII International Conference on Neuroinformatics, October 12-16, 2020, Moscow, Russia. — Springer. 2021. — с. 212—221.

2. HPointLoc: Point-Based Indoor Place Recognition Using Synthetic RGB-D Images / D. Yudin [и др.] // Neural Information Processing: 29th International Conference, ICONIP 2022, Virtual Event, November 22–26, 2022, Proceedings, Part III. — Springer. 2023. — с. 471—484.

3. Real-time object navigation with deep neural networks and hierarchical reinforcement learning / A. Staroverov [и др.] // IEEE Access. — 2020. — т. 8. — с. 195608—195621.

4. *Staroverov A.*, *Panov A. I.* Hierarchical landmark policy optimization for visual indoor navigation // IEEE Access. — 2022. — т. 10. — с. 70447—70455.

5. Skill Fusion in Hybrid Robotic Framework for Visual Object Goal Navigation / A. Staroverov [и др.] // Robotics. — 2023. — т. 12, № 4. — с. 104.

6. *Aleksey S.*, *Panov A. I.* Hierarchical actor-critic with hindsight for mobile robot with continuous state space // Advances in Neural Computation, Machine Learning, and Cognitive Research III: Selected Papers from the XXI International Conference on Neuroinformatics, October 7-11, 2019, Dolgoprudny, Moscow Region, Russia. — Springer. 2020. — с. 62—70.

Список литературы

7. Habitat: A Platform for Embodied AI Research / Manolis Savva* [и др.] // Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). — 2019.

8. A Generalist Agent / S. Reed [и др.]. — 2022. — DOI: 10.48550/ARXIV.2205.06175. — URL: https://arxiv.org/abs/2205.06175.

9. TEACh: Task-driven Embodied Agents that Chat / A. Padmakumar [и др.]. — 2021. — DOI: 10.48550/ARXIV.2110.00534. — URL: https://arxiv.org/abs/2110.00534.

10. AI2-THOR: An Interactive 3D Environment for Visual AI / E. Kolve [и др.]. — 2019. — arXiv: 1712.05474 [cs.CV].

11. Application of pretrained large language models in embodied artificial intelligence // Doklady Mathematics. т. 106. — Springer. 2022. — S85—S90.

12. SegFormer: Simple and efficient design for semantic segmentation with transformers / E. Xie [и др.] // Advances in Neural Information Processing Systems. — 2021. — т. 34. — с. 12077—12090.

13. BlendMask: Top-down meets bottom-up for instance segmentation / H. Chen [и др.] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. — 2020. — с. 8573—8581.

14. SOLOv2: Dynamic, Faster and Stronger / X. Wang [и др.] // arXiv preprint arXiv:2003.10152. — 2020.

15. *Sumikura S.*, *Shibuya M.*, *Sakurada K.* OpenVSLAM: A versatile visual SLAM framework // Proceedings of the 27th ACM International Conference on Multimedia. — 2019. — с. 2292—2295.

16. *Mur-Artal R.*, *Tardos J. D.* ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras // IEEE Transactions on Robotics. — 2017. — окт. — т. 33, № 5. — с. 1255—1262. — DOI: 10.1109/tro.2017.2705103. — URL: http://dx.doi.org/10.1109/TRO.2017.2705103.

17. Using Deep Reinforcement Learning with Automatic Curriculum Learning for Mapless Navigation in Intralogistics / H. Xue [и др.] // Applied Sciences. — 2022. — т. 12, № 6. — DOI: 10.3390/app12063153. — URL: https://www.mdpi.com/2076-3417/12/6/3153.

18. Theta^*: Any-angle path planning on grids / A. Nash [и др.] // AAAI. т. 7. — 2007. — с. 1177—1183.

19. Object Goal Navigation using Goal-Oriented Semantic Exploration / D. S. Chaplot [и др.]. — 2020. — DOI: 10.48550/ARXIV.2007.00643. — URL: https://arxiv.org/abs/2007.00643.

20. Learning to Explore using Active Neural SLAM / D. S. Chaplot [и др.]. — 2020. — DOI: 10.48550/ARXIV.2004.05155. — URL: https://arxiv.org/abs/2004.05155.

21. DD-PPO: Learning Near-Perfect PointGoal Navigators from 2.5 Billion Frames / E. Wijmans [и др.]. — 2019. — DOI: 10.48550/ARXIV.1911.00357. — URL: https://arxiv.org/abs/1911.00357.

22. Auxiliary tasks and exploration enable objectgoal navigation / J. Ye [и др.] // Proceedings of the IEEE/CVF International Conference on Computer Vision. — 2021. — с. 16117—16126.

23. *Sutton R. S.*, *Barto A. G.* Reinforcement learning: An introduction. — MIT press, 2018.

24. Auxiliary Tasks and Exploration Enable ObjectNav / J. Ye [и др.]. — 2021. — arXiv: 2104.04112 [cs.CV].

25. Mastering Atari with Discrete World Models / D. Hafner [и др.]. — 2020. — DOI: 10.48550/ARXIV.2010.02193. — URL: https://arxiv.org/abs/2010.02193.

26. *Rasmussen D.*, *Voelker A.*, *Eliasmith C.* A neural model of hierarchical reinforcement learning // PloS one. — 2017. — т. 12, № 7. — e0180234.

27. *Hengst B.* Hierarchical approaches // Reinforcement Learning: State-of-the-Art. — 2012. — с. 293—323.

28. Large Batch Simulation for Deep Reinforcement Learning / B. Shacklett [и др.] // International Conference On Learning Representations (ICLR). — 2021.

29. *Aleksey S.*, *Panov A. I.* Hierarchical actor-critic with hindsight for mobile robot with continuous state space // Advances in Neural Computation, Machine Learning, and Cognitive Research III: Selected Papers from the XXI International Conference on Neuroinformatics, October 7-11, 2019, Dolgoprudny, Moscow Region, Russia. — Springer. 2020. — с. 62—70.

30. Successor Feature Landmarks for Long-Horizon Goal-Conditioned Reinforcement Learning / C. Hoang [и др.]. — 2021. — arXiv: 2111.09858 [cs.LG].

31. Habitat-Matterport 3D Dataset (HM3D): 1000 Large-scale 3D Environments for Embodied AI / S. K. Ramakrishnan [и др.]. — 2021. — DOI: 10.48550/ARXIV.2109.08238. — URL: https://arxiv.org/abs/2109.08238.

32. Interactive Gibson Benchmark: A Benchmark for Interactive Navigation in Cluttered Environments / F. Xia [и др.] // IEEE Robotics and Automation Letters. — 2020. — апр. — т. 5, № 2. — с. 713—720. — DOI: 10.1109/lra.2020.2965078. — URL: https://doi.org/10.1109%2Flra.2020.2965078.

33. Matterport3D: Learning from RGB-D Data in Indoor Environments / A. Chang [и др.] // International Conference on 3D Vision (3DV). — 2017.

34. Asynchronous Methods for Deep Reinforcement Learning / V. Mnih [и др.]. — 2016. — arXiv: 1602.01783 [cs.LG].

35. Proximal Policy Optimization Algorithms / J. Schulman [и др.]. — 2017. — arXiv: 1707.06347 [cs.LG].

36. *Alatise M. B.*, *Hancke G. P.* A Review on Challenges of Autonomous Mobile Robot and Sensor Fusion Methods // IEEE Access. — 2020. — т. 8. — с. 39830—39846.

37. *Jadidi M. G.*, *Miro J. V.*, *Dissanayake G.* Gaussian processes autonomous mapping and exploration for range-sensing mobile robots // Autonomous Robots. — 2018. — т. 42, № 2. — с. 273—290.

38. *Fang B.*, *Ding J.*, *Wang Z.* Autonomous robotic exploration based on frontier point optimization and multistep path planning // IEEE Access. — 2019. — т. 7. — с. 46104—46113.

39. *Al Khatib E. I.*, *Jaradat M. A. K.*, *Abdel-Hafez M. F.* Low-Cost Reduced Navigation System for Mobile Robot in Indoor/Outdoor Environments // IEEE Access. — 2020. — т. 8. — с. 25014—25026.

40. Mask r-cnn / K. He [и др.] // Proceedings of the IEEE international conference on computer vision. — 2017. — с. 2961—2969.

41. Yolact++: Better real-time instance segmentation / D. Bolya [и др.] // arXiv preprint arXiv:1912.06218. — 2019.

42. The Pascal Visual Object Classes (VOC) Challenge / M. Everingham [и др.] // International Journal of Computer Vision. — 2010. — июнь. — т. 88, № 2. — с. 303—338.

43. From Coarse to Fine: Robust Hierarchical Localization at Large Scale / P.-E. Sarlin [и др.]. — 2018. — arXiv: 1812.03506 [cs.CV].

44. Dueling Network Architectures for Deep Reinforcement Learning / Z. Wang [и др.]. — 2016. — arXiv: 1511.06581 [cs.LG].

45. *Fujimoto S.*, *Hoof H. van*, *Meger D.* Addressing Function Approximation Error in Actor-Critic Methods. — 2018. — arXiv: 1802.09477 [cs.AI].

46. Exploration by Random Network Distillation / Y. Burda [и др.]. — 2018. — arXiv: 1810.12894 [cs.LG].

47. Object Goal Navigation using Goal-Oriented Semantic Exploration / D. S. Chaplot [и др.]. — 2020. — arXiv: 2007.00643 [cs.CV].

48. *Staroverov A.*, *Panov A. I.* Hierarchical landmark policy optimization for visual indoor navigation // IEEE Access. — 2022. — т. 10. — с. 70447—70455.

49. *Muravyev K.*, *Bokovoy A.*, *Yakovlev K.* Enhancing exploration algorithms for navigation with visual SLAM // Russian Conference on Artificial Intelligence. — Springer. 2021. — с. 197—212.

50. *Rozenberszki D.*, *Majdik A. L.* LOL: Lidar-only odometry and localization in 3D point cloud maps // 2020 IEEE International Conference on Robotics and Automation (ICRA). — IEEE. 2020. — с. 4379—4385.

51. *Labbé M.*, *Michaud F.* RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation // Journal of Field Robotics. — 2019. — т. 36, № 2. — с. 416—446.

52. Learning Transferable Visual Models From Natural Language Supervision / A. Radford [и др.]. — 2021. — DOI: 10.48550/ARXIV.2103.00020. — URL: https://arxiv.org/abs/2103.00020.

53. Simple but Effective: CLIP Embeddings for Embodied AI / A. Khandelwal [и др.]. — 2021. — DOI: 10.48550/ARXIV.2111.09888. — URL: https://arxiv.org/abs/2111.09888.

54. ProcTHOR: Large-Scale Embodied AI Using Procedural Generation / M. Deitke [и др.]. — 2022. — DOI: 10.48550/ARXIV.2206.06994. — URL: https://arxiv.org/abs/2206.06994.

55. *Gadzicki K.*, *Khamsehashari R.*, *Zetzsche C.* Early vs late fusion in multimodal convolutional neural networks // 2020 IEEE 23rd International Conference on Information Fusion (FUSION). — IEEE. 2020. — с. 1—6.