Министерство науки и высшего образования Российской Федерации
Московский физико-технический институт
(национальный исследовательский университет)
Физтех-школа прикладной математики и информатики
Магистерская программа
«Методы и технологии искусственного интеллекта»

Осколков Никита Сергеевич

09.04.01 Информатика и вычислительная техника

Литературный обзор для Магистерской диссертации

Научный руководитель:
к.т.н.
Макаров Дмитрий Александрович

Москва
2024 г.

Содержание

1 Введение 3

2 Постановка задачи со стороны обучения с подкрпелением 3

3 Постановка задачи со стороны оптимального управления 4

4 Семантическая навигация на основе обучения с подкреплением мобильного робота с динамическими препятствиями 4

5 Метод с прямым и обратным обучениям с расписанием 6

6 Навигация на основе обучения с подкреплением на основе трансформера 9

7 Метод навигации основанный на графе знаний 11

8 Метод навигации основанный на подзадачах и механизме убеждения 13

Список литературы 16

1 Введение

В последнее собое внимание уделяется разработке колёсных роботов, которые обладают рядом преимуществ перед другими типами роботов. Они отличаются высокой манёвренностью и относительно простой конструкцией. Однако одной из главных проблем при создании колёсных роботов является разработка системы навигации, которая позволила бы им эффективно ориентироваться в пространстве и выполнять поставленные задачи, избегая множества различного рода препятствий.

Подходы с использованием карт устаревают, т.к. чаще всего робот находится в неидеальных условиях с неполными данными об окружающей среде. При навигации без карт мобильный робот перемещается без использования или создания какой-либо карты (например, SLAM). Это реактивная навигация, при которой навигационные решения основаны на информации, получаемой из окружающей среды в режиме онлайн с помощью измерений дальности или датчиков зрения без каких-либо предварительных знаний об окружающей среде. Поскольку мобильный робот использует только текущие данные с датчиков, навигация без отображения на карте в первую очередь решает проблему обхода препятствий. Для навигации без картографирования обычно используются системы на основе датчиков зрения и дальности. Однако использовать зрение для сенсорного наблюдения при навигации роботов сложнее, чем датчики дальности, поскольку отсутствует четкая геометрическая информация.

Для задачи безкартовой навигации по RGB изображению с камеры существует насколько подходов. Обычно в качестве пространства наблюдения используются помимо векторного представления получаемого изображения дополнительные данные, такие как данные глубины, данные с лидара, лазера, компаса (направление до цели), координаты цели в полярных координатах.

2 Постановка задачи со стороны обучения с подкрпелением

Задача алгоритма f навигации мобильного робота состоит в том, чтобы посредствам выбора угловой и линейной скорости в каждый момент движения достичь целевого положения минимизируя затраты на перемещение робота, при этом избегая солкновений с препятствиями.

Процесс взаимодействия с окружающей средой представляет собой марковский процесс принятия решений (MDP), который требует определения непрерывных:

1) пространства состояний $\mathcal{S}$ (State space) -

а) данные с камеры - тензор формата $(200 \times 200 \times 3)$ - X_{cam},

б) вспомогательные данные разного рода (граф знаний, изображение объекта, текст)

c) скорость базы $V_{current}$(2 dof),
(Скорость состоит из двух компонент: линейной и угловой).

2) пространства действий A (Action space)-

скорость $V_{control} \in \overline{V}$, где $\overline{V}$ - класс допустимого управления,

$$V_{control} = f(X_{cam}, X_{gpal}, V_{current}, M_{obstacles})$$

Целью управления является перевод базы из начального положения в положение перед целевым объектом, перпендикулярно столу на котором лежит объект, с ориениеацией направленной на объект с пределенной погрешностью, избегая коллизий с предметами, внутри команты и стенами.

В каждый момент времени состояние робота удовлетворяет $\overline{S}$, где $\overline{S} \subset S$ - класс допустимых (collision-free) состояний базы.

3 Постановка задачи со стороны оптимального управления

В рассматриваемой задаче имеется неподвижная комната G , внутри которой располагаются подвижные окружности-препятствия $\Gamma_i(t), i = 1, ..., N_\Gamma, t \in [t_0, T]$, совершающие плоскопараллелн движения по известным программным законам. Препятствия задают фазовое ограничение, имеющее непустые сечения $\Phi(t) = G/(\bigcup_{i=1}^{N_\Gamma} \Gamma_i(t)) \subset R^2$.

Пусть динамика систмы описываеся уравнением

$$\dot{x} = f(t, x, u) \qquad (1)$$

, где $u(t) = (u_1, u_2)^T$ - управляющее воздействие (скорости).
Требуется построить допустимое управление $u(t), t \in [t_0, T]$, приводящее фазовый вектор $x(t)$ системы (1) из стартового множества X_* на ϵ^*-окрестность целевого множества $X_f(t)$ за минимальное время. Здесь ϵ^* - заданное положительное число.

4 Семантическая навигация на основе обучения с подкреплением мобильного робота с динамическими препятствиями

Вход:
- Данные 360-градусного лазерного сканера;
- расстояние и угол от робота до цели;
- расстояние до динамических объектов (людей и роботов);

Выход:
- Действие робота из следующего списка.

TABLE I: Agent definition

Action	Linear Velocity [m/s]	Angulat Velocity [rad/s]
Forward	0.15	0
Backwards	-0.15	0
Stop	0	0
Left	0.15	+ 0.75
Right	0.15	- 0.75
Strong Left	0.15	+ +1.5
Strong Right	0.15	- 1.5

Рис. 1:
Возможные действия агента

Постановка задачи:
Агент должен добраться до цели поддерживая безопасное расстояние до обнаруженных людей (0.8 м) и сотрудничающих роботов (0.3 м).

Ключевая идея:
В данной работе нам важно, что была успешно использована семантическая иформация. Агент мог раличть робота, человека и статические препятсвия и вести себя с ними требуемым образом.

Алгоритм: DQN
Для модуля обнаружения объектов мы использовали модуль оценки положения, работающий с RGB-входом на базе SSPE(Single Shot Pose Estimation)[1] (основанном на YOLO нейронной сети для оценки позы с 6 степенями свободы, использующий только данные RGB в качестве входных данных). Таким образом, положение и расстояние до людей или сотрудничающих роботов могут быть определены глобально.

В общей сложности было проведено обучение на 3 разных возрастающих по сложности этапах, которые мы обозначаем как статический, динамический и семантический.

Результаты:
Внедрение модели в реального робота прошло без каких-либо трудностей, также было предоставлено сравнение описанного подхода с традиционным локальным планировщиком навигационного пакета Turtlebot3 с точки зрения таких важных показателей, как скорость прохождения дистанции, частота ошибок и безопасность навигации.
Вывод:
В данной работе был изучен потенциал семантической информации: включив семантические классы, такие как человек и робот, была усовершенствована безопасность при навигации. Алгоритм навигации работает исключительно на основе визуального ввода и устраняет необходимость в какой-либо дополнительной карте.

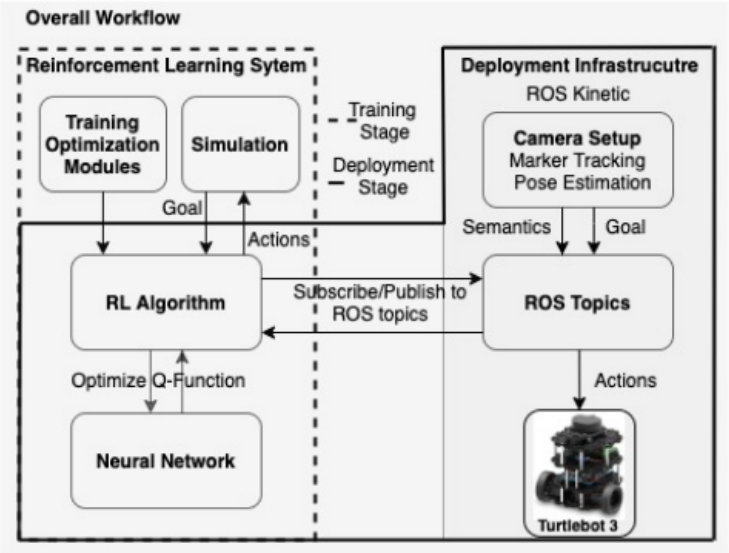


Рис. 2: Дизайн навигационной системы

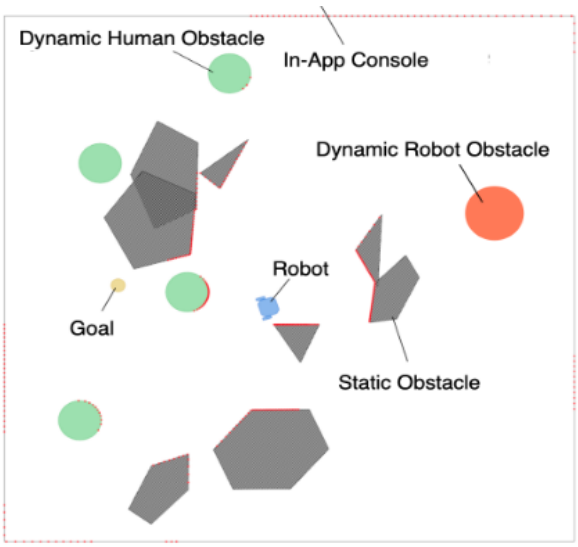


Рис. 3:
Пример симуляции

5 Метод с прямым и обратным обучениям с расписанием

Ключевая проблема, рассмотренная в этой статье, заключается в том, как эффективно и практически преодолеть проблему исследования в обучении с подкреплением (RL), когда доступно всего несколько демонстраций.

Вход:
- Набор данных демонстраций D = {0, 1, ..., tN}, где каждый ti представляет собой траекторию состояний и действий.

Постановка задачи:
Рассматривается стандартная Марковская модель принятия решений (MDP), где агент должен научиться стратегии, максимизирующей ожидаемую дисконтированную награду в среде с редкими наградами (+1 за достижение целевого состояния, 0 иначе)

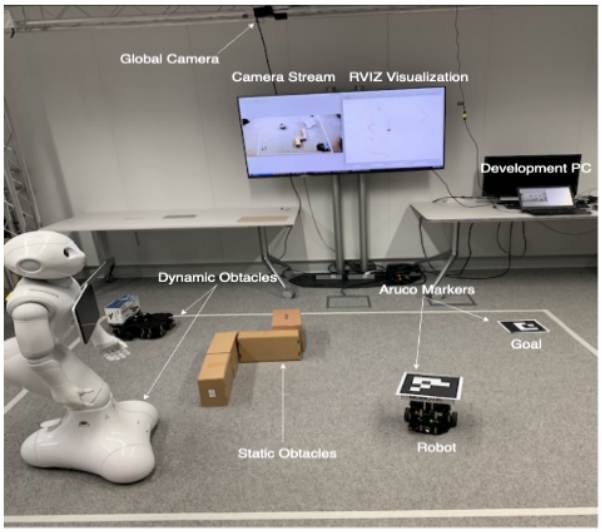


Рис. 4:
Пример работы агента в реальной среде

Metric	Trad.	DRL Stat.	DRL Dyn.	DRL Sem.
Distance [m]	4.72	3.71	**4.1**	5.28
Speed [s]	15.7	**11.7**	12.49	17.9
Error Rate [%]	16.66	10	3.33	**0**
Obstacles hit	6.4	4.8	3.2	**0**

Рис. 5: Сравнение навигационных подходов

Ключевая идея:
Авторы предлагают алгоритм Reverse Forward Curriculum Learning (RFCL) [2], который использует обратное и прямое обучение с расписанием для достижения эффективности по использованию демонстраций и взаимодействий со средой.

Обратнаое обучение с расписанием:
Этот этап план помогает агенту научиться решать задачу на основе узкого распределения начальных состояний, представленного в демонстрациях. Обратный учебный план позволяет преодолеть проблемы с поиском, инициализируя агента вблизи успешных состояний и постепенно удаляя его от них.
Обратное обучение с расписанием разработано таким образом, чтобы использовать демонстрации путем изменения состояний на основе предоставленных траекторий, позволяя первоначальной стратегии эффективно извлекать уроки из узкого распределения состояний.

Прямое обучение с расписанием:
Здесь происходит обобщение первоначальную политику для более широкого распределения начальных состояний. Постепенно отбираются более сложные начальные состояния для обучения без использования дополнительных демонстрационных данных.

Ключевая идея заключается в том, что обратнаое обучение с расписанием помогает агенту преодолеть трудности исследования, в то время как прямое обучение с расписанием

позволяет агенту обобщить приобретенные навыки для выполнения полной задачи. Сочетание этих двух подходов к обучению по учебной программе является основным нововведением метода RFCL.

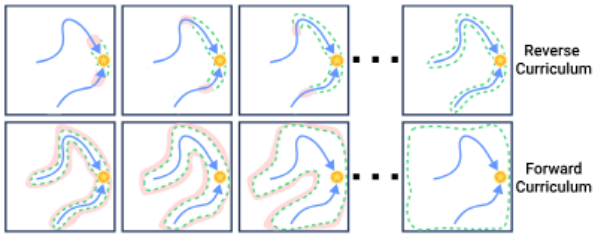


Рис. 6:
Схематичное изображение прямого и обратного обучения с расписанием

Эксперимент:
Эксперименты проводились в 21 полностью наблюдаемой среде с небольшим количеством наград, охватывающей 3 набора роботизированных сред: MetaWorld, Adroit и ManiSkill2. Авторы использовали абсолютно разреженную награду, где +1 начисляется только за успешное выполнение, а 0 - в противном случае. Задания варьировались от простых до сложных.
Также авторы провели игрушечный эксперимент в среде pointmaze с непрерывным пространством состояний/действий, чтобы продемонстрировать влияние обратной и прямой учебных программ, используемых в RFCL. Это показало, как подходы к изучению учебной программы позволяют агенту быстро научиться хорошо справляться с демонстрационным состояниями, а затем обобщать для решения задачи состояния по всему лабиринту.

Результаты:
Результаты представлены на рисунке 7.

Вывод:
RFCL значительно повышает эффективность обучения на основе демонстраций, позволяя алгоритму решать сложные задачи с меньшим количеством демонстраций по сравнению с традиционными методами.
Основная задача, которую решает Goal-guided Transformer (GoT) в предложенном

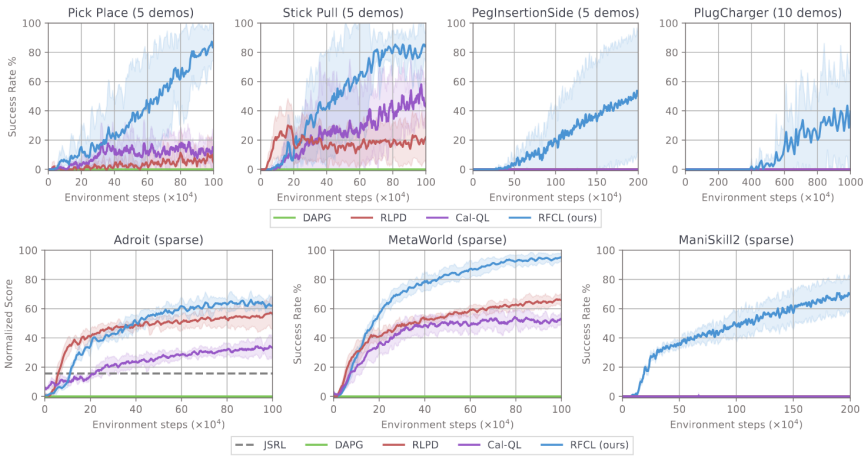


Рис. 7: Результаты эксериментов

подходе Goal-guided Transformer-enabled Reinforcement Learning (GTRL), заключается в том, чтобы сделать представление сцены более ориентированным на цель. А именно GoT рассматривает физическое состояние цели в качестве входа, а не условия, чтобы направлять представление сцены на сопряжение с информацией о цели. Это позволяет представлению сцены сосредоточиться в основном на релевантных для достижения цели признаках, что существенно повышает эффективность обучения последующей системы принятия решений на основе RL.

6 Навигация на основе обучения с подкреплением на основе трансформера

Также существуют работы использующие трансформеры в обучении с подкреплением[3].
Вход:
- 4 самых последних RGB-изображения с камеры "рыбий глаз". Входные изображения также дополнены гауссовым шумом для повышения надежности и приближения к реальным. - Положение цели в двумерной полярной системе координат, представленное относительным расстоянием и углом направления к цели.

Выход:
- Непрерывные управляющие воздействия: линейную и угловую скорости.

Постановка задачи:
Основная цель автономной навигации, достижение заданной целевой позиции в неизвестной среде без использования карты.

Ключевая идея:

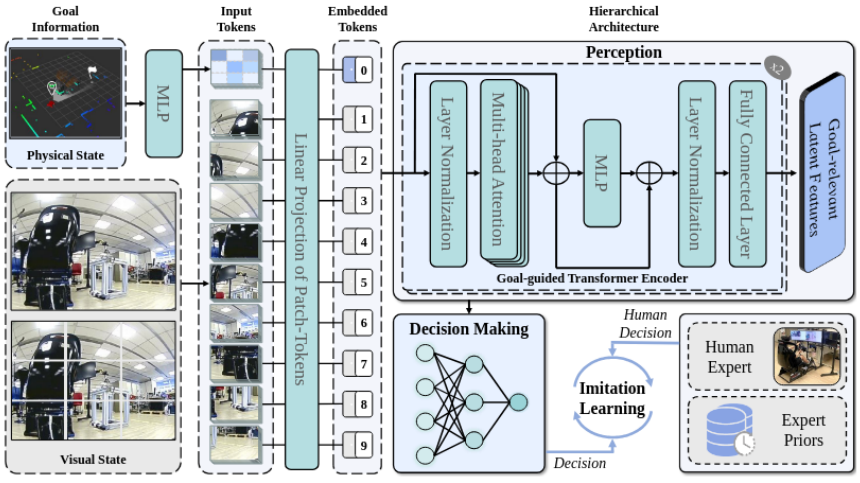
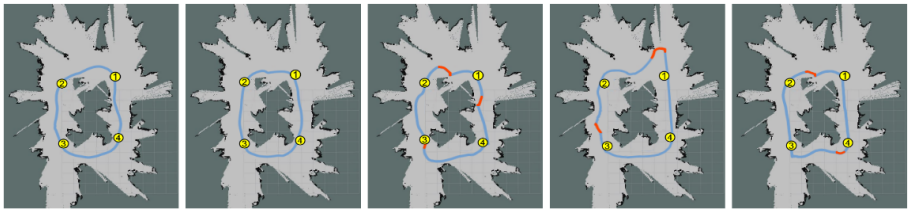


Рис. 8:
Общая схема подхода

Ключевые этапы алгоритма, описанного в статье, следующие:



(a) GoT-SAC. (b) ViT-SAC. (c) ConvNet-SAC. (d) MultiModal CIL. (e) MoveBase.

Рис. 9: Траектории внутри эксперимента. Красным выделена помощь человека

1. Использование Goal-Guided Transformer (GoT) для извлечения целевых признаков из входных данных: Алгоритм использует GoT в качестве основы системы восприятия. GoT объединяет информацию о целевой позиции и визуальные данные в единое представление.

2. Предварительное обучение GoT с использованием демонстрационных данных: Перед основным обучением, GoT предварительно обучается на демонстрационных данных с использованием имитационного обучения. Это позволяет повысить эффективность обучения последующей системы принятия решений.

3. Обучение системы принятия решений на основе алгоритма Soft Actor-Critic (SAC): Извлеченные GoT целевые признаки используются в качестве входных данных для алгоритма SAC, который обучается принимать решения для навигации к целевой позиции.

Ключевой особенностью алгоритма является использование GoT: GoT преобразует многомодальный вход (физическое состояние цели и визуальное состояние) в компактное цель-ориентированное представление, которое фокусируется на релевантных для достижения цели признаках.

Эксперимент:
Участники эксперимента выполняли демонстрацию задачи навигации к заданной цели, управляя автомобилем с помощью педалей и рулевого колеса. Всего было собрано 200 траекторий, состоящих из 17215 пар состояние-действие.
Были протетсированны самые современные алгоритмы в качестве базовых для сравнения качественных и количественных характеристик как в симуляторе, так и в реальном мире. Нужно было достич последовательно 4 позиций 9.
Результаты:

GoT-SAC обеспечивает плавную навигацию без столкновений, в то время как три других алгоритма (ConvNet-SAC, мультимодальный CIL и Move Base) требуют участия человека для прибытия в пункты назначения.

Вывод:
В целом, эксперименты продемонстрировали, что предлагаемый подход GTRL превосходит другие методы по эффективности обучения, производительности, устойчивости и способности

Approach	Goal Position	Avg. Dist.	Var. Dist.	Avg. Time.	Var. Time	Success Rate	Engage Number
GoT-SAC	1st	6.044	± 0.049	14.048	± 0.282	100%	0
	2nd	**5.747**	± **0.110**	**11.987**	± **0.368**	100%	0
	3rd	**6.171**	± **0.036**	**12.821**	± **0.076**	100%	0
	4th	6.360	± 0.113	13.902	± 1.287	100%	0
ViT-SAC	1st	5.958	± 0.036	15.274	± 0.394	100%	0
	2nd	6.506	± 0.189	15.432	± 0.741	100%	0
	3rd	6.226	± 0.015	22.502	± 1.601	100%	0
	4th	7.274	± 0.210	16.449	± 0.722	100%	0
ConvNet-SAC[11]	1st	6.150	± 0.294	17.918	± 2.563	100%	4
	2nd	6.780	± 0.252	20.735	± 1.519	100%	5
	3rd	6.322	± 0.245	18.757	± 2.683	100%	6
	4th	5.971	± 0.123	13.550	± 0.414	100%	0
MultiModal CIL[44]	1st	5.852	± 0.021	13.458	± 0.180	100%	0
	2nd	9.868	± 0.480	57.263	± 6.026	60%	5
	3rd	6.666	± 0.134	75.121	± 12.019	20%	5
	4th	6.913	± 0.217	28.893	± 4.475	100%	0
MoveBase	1st	5.822	± 0.063	12.180	± 0.088	100%	0
	2nd	7.070	± 0.529	23.446	± 9.388	100%	4
	3rd	6.092	± 0.141	14.010	± 3.122	100%	1
	4th	6.510	± 0.517	24.098	± 5.442	100%	5

Рис. 10: Сравнительная эффективность предложенного алгоритма по сравнению с базовыми показателями. Данные, выделенные жирным шрифтом, указывают на лучшие результаты

к переносу в реальный мир.

7 Метод навигации основанный на графе знаний

Данная работа [4] выделяется тем, что в ней реализовано графовое представление, в котором узлы представляют семантические местоположения, а ребра - навигационное поведение, что позволяет роботам работать на более высоком уровне абстракции без явных геометрических вычислений. Этот метод использует архитектуру глубокого обучения для прямого преобразования визуальных данных в средства управления двигателем, повышая способность робота ориентироваться в условиях, когда GPS недоступен.

Вход:
- RGB изображение
- вектор, который предоставляет дополнительную информацию о подцелях, связанных с текущим выполнением действия (например о выходе из зала через левый выход).

Выход:
- 2-ух мерный ветор скорости по осям x и y, привязанным к роботу, и угловая скорость.

Постановка задачи:
Агенту необходимо доехать из одной точки местности до друой, где местность состоит из комнат, коридоров и опознавательных объектов (картин на стенах).

Ключевая идея:
В работе представлены два основных подхода к поведению для автономной навигации в

Code	Description
ool	out of office, take left
oor	out of office, take right
cf	follow-corridor
iol	enter office to the left
ior	enter office to the right
chs	cross hall, continue straight
chl	cross hall, take left
chr	cross hall, take right
ccc	change corridor, straight
ccl	change corridor to the left
ccr	change corridor to the right

Рис. 11: Закодированные действия

помещениях:
Реактивные поведения (Reactive behaviors):
- Представляют собой прямое отображение от восприятия к действию, без использования явного внутреннего представления мира.
- Реализуют непосредственное управление роботом на основе визуального ввода.

Поведения на основе памяти (Memory-based behaviors):
- Используют явное внутреннее представление мира для планирования последовательности действий.
Поведение, основанное на запоминании, в этом подходе использует явное внутреннее представление или память для распознавания ранее посещенных мест в окружающей среде. Ключевыми аспектами реализации памяти являются:
Определение ориентиров: окружающая среда ассоциируется с определенными визуальными ориентирами, такими как картины, висящие на стенах.
Во время обучения набор визуальных описаний (карты объектов) извлекается из нескольких изображений, содержащих каждый ориентир, с использованием предварительно обученной светрчной нейронной сети. Полученные дескрипторы затем передаются через обучаемую билинейную функцию для получения вложений для каждого ориентира. Вложения сохраняются в виде ключей в модуле памяти с соответствующим идентификатором места в качестве адресуемого значения. Идентификатор места по умолчанию также используется для обработки случаев, когда входное изображение не содержит действительного визуального ориентира. Далее используется косинусоидальное расстояние для оценки сходства между дескриптором каждой области изображения и кодировкой визуальных ориентиров, хранящихся в памяти. Граф может интерпритироватся как триплеты формы ⟨место|поведение|место⟩

Пример работы: агента когда робот следует по коридору, обнаружение определенного ориентира может сигнализировать о необходимости переключения на режим входа в офис.

Эксперимент:
Возможные действия:
1) Следование по коридору (cf): реактивное навигационное поведение , которое указывает роботу путь при его перемещении по коридору. Входными данными для этого поведения является текущий вид робота (RGB-изображение).
2) Out-Office (oo)
3) Вход в офис (io)
4) Переход через зал (ch):
5) Изменение коридора (cc):

Navigational Behaviors

	Train: #paths	Train: #images	Test: #trials	Accuracy
cf	150	46131	100	100%
io	8000	221734	100	100%
oo	8000	259937	100	67%
oo-b	8000	259937	100	96%
ch	3000	248950	100	100%
cc	4000	123653	100	100%

Perceptual Behaviors

	-	Train: #images	Test: #images	Class. Acc.
pd	-	12945	1500 (Imgs)	98.2%
lmpd	-	38459	1500 (Imgs)	96.7%

Рис. 12: Результаты

6) Определение местоположения (pd): функция восприятия, которая определяет тип места, где в данный момент находится робот. Возможные категории: офис, коридор, холл.

Результаты:
Результаты представлены в таблице 12. Точность выполнения навигационных поведений составила от 67% до 100% в зависимости от типа поведения. При выполнении миссий по перемещению между офисами, робот успешно достигал цели в 81,2% случаев в среднем за 8,3 активации навигационных поведений.
Основные причины ошибок - неточность распознавания ориентиров и сложность выхода из офиса при неудачном начальном положении робота.

Использование набора поведенческих навыков (navigational behaviors), которые напрямую управляют двигателями робота на основе визуальных входных данных, без необходимости явного вычисления местоположения робота или геометрии окружающей среды.

Представление среды в виде графа, где узлы соответствуют семантическим местам (офисы, коридоры, залы), а ребра - навигационным поведениям (войти в офис, пересечь коридор и т.д.).

Вывод:
Авторы отмечают, что предложенный подход предлагает альтернативу традиционным методам навигации, основанным на локализации, и обеспечивает более компактное семантическое представление среды, что может облегчить взаимодействие с человеком и применение методов планирования.

8 Метод навигации основанный на подзадачах и механизме убеждения

Вход:
- Визуальные входные данные (RGBD изображения)
- Семантическая сегментация визуальных входных данных
- Датчик GPS+компас, предоставляющий информацию о местоположении и ориентации

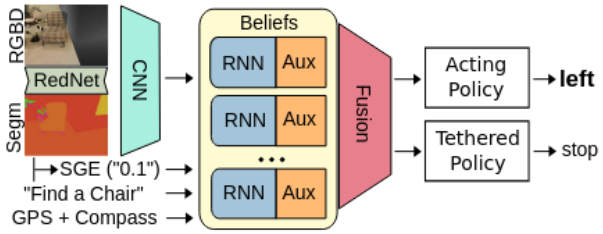


Рис. 13: Архитектура подхода

агента
- Идентификатор целевого объекта категории, который агент должен найти

Выход:
- Действия агента из дискретного набора: MOVE_FORWARD, TURN_LEFT, TURN_RIGHT, LOOK_UP, LOOK_DOWN, STOP
Постановка задачи:
Агенту необходимо найти и достичь объекта определенной категории в незнакомой окружающей среде.

Ключевая идея:
Авторы [6] предложили использовать две стратегии - одну, оптимизированную с помощью поощрения за исследование, и вторую, оптимизированную только с помощью целевого вознаграждения за достижение объекта. Это позволило агенту сначала научиться эффективно исследовать, а затем быстро адаптироваться к целевой задаче.
Ключевым нововведением являются модули убеждения 13:
Модули убеждений - это независимые GRU (Gated Recurrent Unit), каждый из которых связан с отдельной вспомогательной задачей. Они используются для преобразования наблюдений во временные представления, называемые "убеждениями".

Идея состоит в том, что разделение модулей на независимые GRU, а не использование одной большой рекуррентной сети, позволяет обучать ортогональные вспомогательные задачи, минимизируя конфликт между ними.

Эксперимент:
Эксперименты проводились на датасете Matterport (MP3D), который содержит 90 сцен и 40 помеченных семантических категорий объектов. Из них 21 категория использовалась в качестве целевых.
Результаты:
ВИспользование впомогательных задач обеспечивают умеренный прирост производительности. Обучение с привязкой лучше сохраняет показатель успеха, чем прямая настройка на разреженное вознаграждение за достижение целевых положений.

Вывод:
В целом, работа демонстрирует, что с помощью вспомогательных задач, поощрения за исследование и использования семантической информации можно обучить эффективного агента для навигации к объектам, не полагаясь на явные пространственные карты окружения.

Было показано, что существует ряд подходов, которые могут помочь решить задачу безкартовой навигации при неполных данных.

Список литературы

[1] Kästner, Linh, Cornelius Marx, and Jens Lambrecht. "Deep-reinforcement-learning-based semantic navigation of mobile robots in dynamic environments."2020 IEEE 16th International Conference on Automation Science and Engineering (CASE). IEEE, 2020.

[2] Tao, Stone, et al. "Reverse Forward Curriculum Learning for Extreme Sample and Demo Efficiency."The Twelfth International Conference on Learning Representations.

[3] Huang, Wenhui, et al. "Goal-guided transformer-enabled reinforcement learning for efficient autonomous navigation."IEEE Transactions on Intelligent Transportation Systems (2023).

[4] Sepulveda, Gabriel, Juan Carlos Niebles, and Alvaro Soto. "A deep learning based behavioral approach to indoor autonomous navigation."2018 IEEE international conference on robotics and automation (ICRA). IEEE, 2018.

[5] Ye, Joel, et al. "Auxiliary tasks and exploration enable objectgoal navigation."Proceedings of the IEEE/CVF international conference on computer vision. 2021.