

ПОСТРОЕНИЕ КОМПАКТНЫХ ГРАФОВ СЦЕНЫ НА ОСНОВЕ ТОПОЛОГИЧЕСКОЙ КАРТЫ ДЛЯ АВТОНОМНОЙ НАВИГАЦИИ МОБИЛЬНОГО РОБОТА

Муравьев К.Ф.^{1, а}, Романенко В.И.^{2, б}

¹ Федеральный исследовательский центр «Информатика и управление» Российской академии наук (ФИЦ ИУ РАН), Москва, Россия.

² Национальный исследовательский университет «Высшая школа экономики» (НИУ ВШЭ), Москва, Россия.

E-mail: ^а kmuravev@frccsc.ru, ^б viktorsamr@yandex.ru

Аннотация. Автономная навигация мобильного робота в человеко-ориентированных средах требует наличия карты, которая является не только геометрической моделью среды для планирования маршрутов, но также содержит информацию об объектах среды (двери, мебель, офисная техника и т.д.). Такое представление карты обеспечивают графы сцены, в которых вершинами являются помещения, локации и объекты, а ребра кодируют пространственные связности или отношения между объектами. Большинство современных методов построения графов сцены обладают высокой вычислительной сложностью, и построенные ими графы являются избыточными для обеспечения автономной навигации робота. В данной работе предлагается метод построения компактных графов сцены (Compact Scene Graph, CSG), в основе которого лежит вычислительно эффективный метод топологического картирования PRISM-ТороМар и привязка семантических объектов к локациям топологической карты. Построенный граф сцены позволяет строить маршруты до объектов по локациям топологической карты и осуществлять локализацию с высокой точностью. Предложенный метод был экспериментально исследован в симуляционной среде Habitat. Результаты экспериментов показали, что предложенный CSG потребляет значительно меньше памяти, чем традиционные метрические карты и графы сцен, и обеспечивает надежную локализацию по топологической карте за счет привязки к семантическим объектам.

Ключевые слова: *граф сцены, топологическая карта, семантическая сегментация, автономная навигация, симуляционная среда.*

Введение

Автономная навигация является одним из ключевых аспектов автономного функционирования мобильного робота. Способность самостоятельно перемещаться в незнакомой или динамически изменяющейся среде без участия оператора определяет практическую ценность робототехнических систем в самых разных областях – от сервисной робототехники до промышленной автоматизации. Для реализации автономной навигации роботу требуется карта окружающего пространства, которая служит основой для планирования маршрутов, локализации и принятия решений о движении.

Для цитирования:

Муравьев К.Ф., Романенко В.И. Построение компактных графов сцены на основе топологической карты для автономной навигации мобильного робота. *Интеллектуальный транспорт*. 2026. № 3(39). С. 37–52. DOI: 10.24412/3033-6007-2026-339-37-52.

В зависимости от способа представления среды различают несколько типов карт. Метрические карты, такие как сетки занятости или трёхмерные облака точек, обеспечивают высокую геометрическую точность и детализацию, однако потребляют значительный объём памяти и подвержены накоплению ошибок по мере движения робота. Примерами наиболее распространенных метрических подходов к картированию являются RTAB-Map [1] и GLIM [2]. Такие методы удобны тем, что строят плотное геометрическое представление среды и комплексировать данные с разных сенсоров робота, однако из-за высокого потребления памяти и накопления ошибки они пригодны лишь для сред небольшой площади, что подтверждается экспериментами, проведенными в работах [3] и [4].

Альтернативой выступают топологические карты, которые представляют среду в виде графа, где вершины соответствуют локациям среды, а рёбра – пространственной связности между ними. Такие карты компактны и эффективны для планирования маршрутов, однако зачастую лишены структурной детализации и семантической информации, что снижает их различительную способность и затрудняет локализацию в визуально монотонных средах. К числу топологических методов относятся PRISM-ТороМар [5], строящий компактный граф локаций на основе распознавания мест и сопоставления сканов, а также TSGM [6], интегрирующий семантическую информацию в топологическое представление для навигации по целевому изображению. Метод PRISM-ТороМар строит компактные карты сцен большой площади, пригодные для долгосрочной навигации, однако без добавления семантической информации навигация возможна лишь по заданным точным координатам внутри локации. Метод TSGM позволяет задавать цель навигации в виде изображения, однако не предназначен для долгосрочной навигации и подвержен ошибкам распознавания локации.

Свойства метрических и топологических карт объединяют графы сцены – представления, сочетающие геометрическую, семантическую и топологическую информацию об окружающей среде. В таких графах вершины могут соответствовать помещениям, локациям и отдельным объектам, а рёбра кодируют пространственные или семантические отношения между ними. Графы сцены обеспечивают богатое структурированное описание среды, пригодное для высокоуровневого понимания сцены, социально-ориентированной навигации и семантических рассуждений. Однако существующие подходы к их построению, как правило, предназначены для полного понимания сцены (вплоть до отношений вида «кружка стоит на столе»), а не для навигации [7]. В результате построенные ими графы остаются плотными и громоздкими, их построение требует больших вычислительных затрат, и их поддержание на роботизированных платформах с ограниченными ресурсами затруднительно (см. рис. 1). Среди таких методов можно выделить Hydra [8] – систему для построения иерархических трёхмерных графов сцен, объединяющую метрические сетки с семантической информацией, и Khronos [9] – расширение Hydra для динамических сред с моделированием временной размерности.

В данной работе предлагается метод построения компактного графа сцены (Compact Scene Graph, CSG) – промежуточного представления, оптимизированного для автономной навигации мобильного робота в крупных помещениях. Основная идея заключается в использовании вычислительно эффективного метода топологического картирования PRISM-ТороМар в

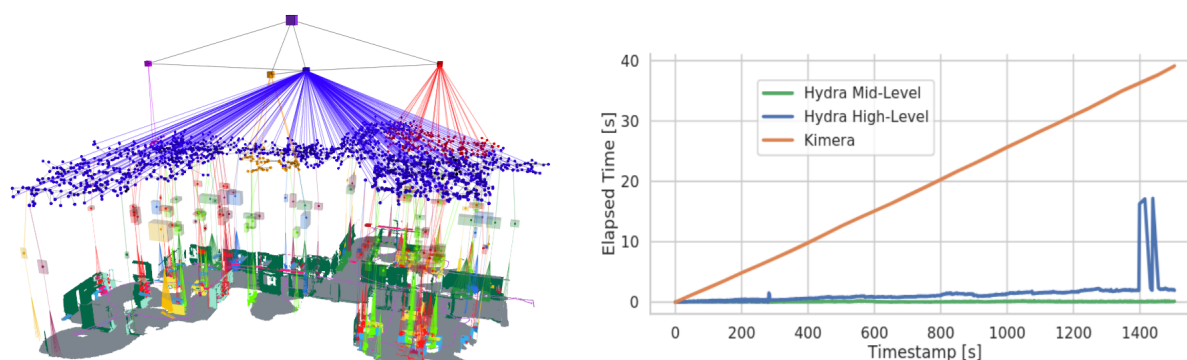


Рис. 1. Граф сцены, построенный методом Hydra (слева), и график времени обновления карты (справа). Граф для представленного помещения занимает 200 МБ, на графике виден линейный рост времени обновления карты. Источник [8]

качестве структурной основы, что обеспечивает компактность и масштабируемость графа. Навигационные вершины, соответствующие посещённым локациям, дополняются облегчённым семантическим слоем: обнаруженные объекты (двери, мебель, офисная техника и т.д.) привязываются к локациям топологической карты, с которых они наблюдались, с обязательной дедупликацией – каждый физический объект представлен ровно одной вершиной, даже если он виден с нескольких локаций. Такое представление позволяет использовать семантическую информацию для переранжирования кандидатов на этапе локализации, повышая её надёжность в визуально неразличимых областях, при этом потребление памяти на хранение и поддержание карты остается значительно более низким, чем у метрических методов картирования и традиционных методов построения графов сцен. Для локализации по построенному CSG в данной работе предлагается использовать коэффициент Жаккара, IDF-взвешивание (IDF – Inverse Document Frequency) и позиционно-осознанный режим.

Экспериментальная оценка предложенного метода проводилась в симуляторе Habitat [10] на сценах, моделирующих помещения большой площади. Результаты показали, что CSG потребляет значительно меньше памяти, чем плотные метрические карты, и превосходит полностью топологические подходы по надёжности локализации в визуально монотонной среде большой площади. Использование семантического переранжирования снизило среднюю ошибку локализации с 2,10 м до 1,20 м при сохранении компактности графового представления. Полученные результаты подтверждают эффективность предложенного подхода как компромисса между геометрической точностью, семантической насыщенностью и вычислительной эффективностью.

1. Обзор литературы

1.1. Методы топологического картирования

Топологические методы представляют среду в виде графа, где вершины соответствуют локациям, а рёбра – пространственной связности между ними. Такие подходы обеспечивают компактность представления и эффективность планирования маршрутов, однако зачастую не содержат семантической информации, необходимой для надёжной локализации. Одним из наиболее эффективных онлайн-методов в этой области является PRISM-ТороМар [5], который строит граф локаций с помощью обучаемого мультимодального

модуля распознавания мест и сопоставления сканов, не полагаясь на точные глобальные координаты робота и поддерживая компактность графа. В работе [6] предлагается топологический семантический граф памяти (TSGM), адаптированный для навигации до целевого изображения; этот метод объединяет структуру обучаемой памяти с навигационными политиками, однако может порождать рёбра между несмежными местоположениями, что снижает его надёжность. Другой подход [11] реализует модульную систему автономного исследования и семантического картирования крупных помещений, в которой на основе построенной метрической карты занятости формируется топологический граф объектов, обновляемый при последующих обходах. Помимо перечисленных, существуют методы, основанные на адаптивной резонансной теории (Adaptive Resonance Theory [12]), иерархическом обучении с подкреплением (ToroNav [13]) и семантическом обнаружении циклов (SemanticToroLoop [14]), которые расширяют возможности топологических карт в задачах долговременной навигации и локализации в сложных условиях.

1.2. Методы построения графов сцены

На настоящий момент разработано множество методов построения графов сцен, которые различаются по целевой задаче, вычислительной сложности и области применения. В зависимости от режима работы их можно разделить на офлайн-методы, работающие с уже реконструированными данными (например, облаками точек и видеопоследовательностями), и онлайн-методы, выполняющие построение графа в реальном времени по мере поступления сенсорной информации.

Офлайн-методы предназначены для обработки заранее собранных данных, таких как трёхмерные реконструкции помещений или облака точек, и, как правило, обеспечивают высокое качество построения графа за счёт возможности использования глобальной информации. Одним из первых подходов в этом направлении стала работа [15], в которой авторы предлагают алгоритм, строящий семантические графы сцен из реконструированных трёхмерных облаков точек с использованием обучаемых признаков объектов и графовых нейронных сетей для предсказания отношений между ними. Альтернативную стратегию предлагает метод [16], рассматривающий рёбра как первичные сущности: здесь используется рёберно-центрированная агрегация для итеративного уточнения предсказаний отношений, что позволяет повысить согласованность построенного графа по сравнению с вершинно-центрированными архитектурами. Помимо указанных работ, в последние годы появилось множество других офлайн-подходов, основанных на трансформерных архитектурах (например, SGFormer++ [17]), кросс-модальном выравнивании с использованием CLIP [18], а также самоконтролируемом обучении (SGRec3D [19]) и языковых онтологиях. Все эти методы демонстрируют высокую точность на эталонных датасетах, однако их вычислительная сложность и зависимость от качества входной реконструкции ограничивают применение в реальных робототехнических системах.

В отличие от офлайн-подходов, онлайн-методы ориентированы на построение графов сцены в реальном времени по мере движения робота, что делает их пригодными для использования на бортовых вычислительных системах. Система Hydra [8] строит иерархический трёхмерный граф,

включающий здания, комнаты, места, объекты и метрическую карту в виде трёхмерной полигональной сетки (меша), объединяя геометрию и семантику в единое структурированное представление. Расширением этой идеи является Khronos [9], который адаптирует построение графов для динамических сред, моделируя фоновую геометрию, фрагменты объектов, полные объекты и позиции робота в четырёх измерениях (пространство и время). Графы сцен, построенные подобными методами, содержат полную информацию о сцене, однако являются очень громоздкими (например, граф одного этажа здания, построенный методом Hydra, занимает 200 МБ памяти при хранении). Такие методы подходят только для картирования небольших помещений, поскольку, как показано авторами метода Hydra, время на одну итерацию обновления графа растет линейно со временем работы системы.

В отличие от перечисленных систем, строящих полное описание сцены, метод Clío [7] строит граф, ориентированный на конкретную задачу: используя текстовое описание задач, система сохраняет только те объекты и области, которые релевантны для достижения поставленной цели, что позволяет значительно сократить размер карты и вычислительные затраты. К числу других онлайн-подходов относятся vS-Graphs [20], S-Graphs+ [21] и ConceptGraphs [22], которые интегрируют понимание сцены непосредственно в процесс картирования и обеспечивают совместную оптимизацию метрических, топологических и семантических уровней представления. Методы vS-Graphs и S-Graphs+ строят компактное описание сцены с элементами семантической информации (определение стен помещений), однако в графе отсутствуют многие полезные семантические классы (двери, элементы мебели). Метод ConceptGraphs, напротив, дает подробное описание сцены с логическими связями, однако для построения этих связей нужны большие языковые модели, использование которых на реальных роботах затруднительно ввиду ограниченности вычислительных ресурсов.

Таким образом, с учетом ограничений всех вышеперечисленных методов, в рамках данной работы был разработан новый метод построения компактных, но семантически информативных графов сцены путем добавления информации о семантических объектах в топологический метод PRISM-ТороМар.

2. Постановка задачи

В данной работе рассматривается задача построения компактного графа сцены, пригодного для локализации мобильного робота в помещениях, по данным с бортовых датчиков робота. Предполагается, что на первом этапе робот строит граф сцены в реальном времени при проезде по заранее заданному маршруту, охватывающему всю доступную для навигации среду. Далее робот осуществляет навигацию в среде, локализуясь с помощью построенного графа, без использования внешних источников позиционирования и глобальных метрических карт.

На каждом шаге t доступны следующие данные с датчиков робота:

- $O_t = (I_t, D_t)$ – **наблюдение**, состоящее из панорамного изображения (угол обзора 360°) и его карты глубин;
- $\Delta p_t \in SE(3)$ – **одометрия** (оценка перемещения робота от шага $t - 1$ к

шагу t).

Требуется разработать алгоритм *Mapping*, который по таким данным будет строить граф $G = (V, E)$.

Граф $G = (V, E)$ содержит два типа вершин и ребер:

- $V = V_{nav} \cup V_{sem}$, где V_{nav} – это множество **навигационных вершин** (локаций), V_{sem} – множество вершин, соответствующих семантическим объектам. Каждая локация $v \in V_{nav}$ имеет свою систему координат с началом координат (v^x, v^y) и углом поворота v^θ в глобальной системе координат. Каждый **семантический объект** $w \in V_{sem}$ задается координатами центра объекта в глобальной системе координат и меткой класса w^c .
- $E = E_{nav} \cup E_{sem}$, где ребра $(v_i, v_j) \in E_{nav}$ связывают пространственно смежные локации из V_{nav} , а ребра $(v_i, w_j) \in E_{sem}$ связывают объект w_j из V_{sem} с локацией v_i из V_{nav} , с которой он был обнаружен. Один и тот же семантический объект может быть связан ребрами с несколькими локациями.

Алгоритм должен работать в реальном времени, т. е. на каждом шаге t выдавать обновленный граф G_t :

$$Mapping(G_{t-1}, O_t, \Delta p_t) = G_t. \quad (1)$$

Также требуется разработать алгоритм локализации *Loc*, который по графу G и данным с робота $(O_t, \Delta p_t)$ будет определять позицию робота в графе – локацию v_{cur}^t , в которой находится робот в момент t , и положение робота $\delta_{cur}^t = (\delta x, \delta y, \delta \theta)$ в системе координат локации:

$$Loc(G, O_t, \Delta p_t) = (v_{cur}^t, \delta_{cur}^t). \quad (2)$$

Целевой метрикой является ошибка локализации – расхождение между истинным положением робота и положением робота, определенным алгоритмом локализации и переведенным в глобальную систему координат:

$$error = \left\| \begin{pmatrix} x_t \\ y_t \end{pmatrix} - \begin{pmatrix} x_{cur} + \cos(\theta_{cur}) \delta_x + \sin(\theta_{cur}) \delta_y \\ y_{cur} - \sin(\theta_{cur}) \delta_x + \cos(\theta_{cur}) \delta_y \end{pmatrix} \right\|, \quad (3)$$

где (x_t, y_t) – истинные координаты робота в момент t ; $(x_{cur}, y_{cur}, \theta_{cur})$ – координаты и угол ориентации локации v_{cur}^t .

3. Предложенный метод построения компактного графа сцены

Предложенный метод в реальном времени строит граф сцены $G = (V, E)$, состоящий из навигационного подграфа (V_{nav}, E_{nav}) и семантического слоя (V_{sem}, E_{sem}) . Навигация и планирование маршрутов осуществляются по навигационному подграфу. Семантический слой задействуется на этапе локализации, а также может быть использован для задания целевой точки для навигации. Пример построенного графа с семантическим слоем приведен на рисунке 2.

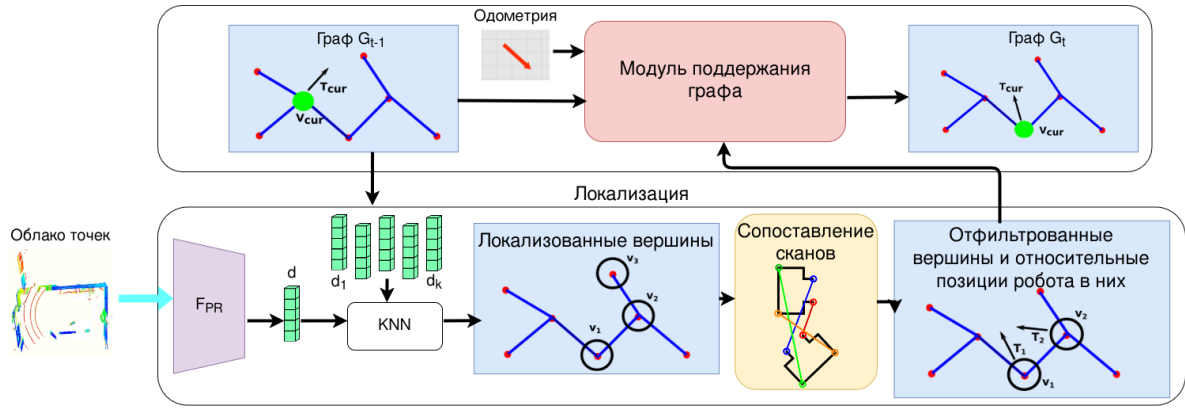


Рис. 3. Схема метода построения навигационного графа PRISM-ТороМар. Источник [5]

3.2. Семантический слой

Для создания графа сцены к навигационному графу, который строится методом PRISM-ТороМар, добавляется семантический слой. Найденные семантические объекты соединяются ребрами с локациями, из которых они были обнаружены. Для обнаружения семантических объектов используется нейросетевая модель GroundingDINO [23]. На вход модели подаются части панорамного изображения с камер робота (4 изображения с охватом 90 градусов каждое) и текстовый запрос, задающий перечень семантических классов. Модель возвращает ограничивающие прямоугольники объектов на изображении, текстовые метки классов и оценки уверенности. Для каждого обнаруженного объекта определяется его положение в пространстве – вычисляется медианная глубина в квадрате размером 5×5 пикселей вокруг центра ограничивающего прямоугольника, далее по глубине и положению на изображении вычисляется положение центра объекта в системе координат робота, которое затем переводится в глобальную систему координат.

Для исключения повторного добавления объектов в граф проводится дедупликация следующим образом. После вычисления глобальных координат объекта (w_x, w_y) проводится поиск ближайшего объекта того же класса $w_{nearest}$ в семантическом подграфе V_{sem} . Если он найден и расстояние до него не превышает порог r_{dup} , то новый семантический объект не создается, а найденный объект $w_{nearest}$ соединяется ребром с локацией v_{cur} . В противном случае создается новый семантический объект w_{new} с координатами (w_x, w_y) , добавляется в семантический подграф и соединяется с локацией v_{cur} .

3.3. Локализация в графе сцены

Модуль локализации на каждом шаге определяет вершину $v_{cur} \in V_{nav}$, в которой находится робот, и его относительное положение $\delta_{cur}^t = (\delta x, \delta y, \delta \theta)$ внутри локации v_{cur} . Локализация проводится в два этапа; первый этап совпадает с базовым методом PRISM-ТороМар. В ходе первого этапа отбираются k локаций-кандидатов с помощью метода распознавания мест. Далее найденные кандидаты переупорядочиваются с использованием объединенной оценки:

$$score(v_i) = (1 - \alpha) (-\|f_{cur} - f_{v_i}\|_2) + \alpha s(v_i), \quad (4)$$

где $\alpha \in [0, 1]$ – вес семантической составляющей, f_{cur} – дескриптор текущего наблюдения, f_{v_i} – дескриптор вершины v_i , $s(v_i)$ – семантическая оценка кандидата. Оценка вычисляется только для тех кандидатов, для которых было успешно найдено сопоставление сканов с помощью базового метода PRISM-ТороМар. Для вычисления величины $s(v_i)$ применяется несколько способов:

Коэффициент Жаккара (jaccard):

$$s(v_i) = J(C_{cur}, C_{v_i}) = \frac{|C_{cur} \cap C_{v_i}|}{|C_{cur} \cup C_{v_i}|}, \quad (5)$$

где C_{cur} – множество классов объектов, обнаруженных в текущем кадре, C_{v_i} – множество классов объектов, наблюдаемых из вершины v_i .

IDF-взвешенный коэффициент Жаккара (idf):

$$idf(c) = \log \frac{N + 1}{df(c) + 1}, \quad (6)$$

где N – число вершин графа, $df(c)$ – число вершин, содержащих хотя бы один объект класса c . Семантическая оценка вычисляется как взвешенный коэффициент Жаккара:

$$s(v_i) = \frac{\sum_{c \in C_{cur} \cap C_{v_i}} idf(c)}{\sum_{c \in C_{cur} \cup C_{v_i}} idf(c)}. \quad (7)$$

Позиционно-осознанный режим (position):

Для вычисления семантической оценки в данном режиме учитываются не только классы обнаруженных объектов, но и их координаты относительно робота. Для каждого объекта, обнаруженного в текущем кадре, зададим тройку (c, o_x, o_y) – семантический класс объекта и его координаты относительно робота. При рассмотрении вершины-кандидата v_i все семантические объекты из V_{sem} , соединенные с ней, переводятся в систему координат робота. Объект с текущего наблюдения $d = (c, o_x, o_y)$ считается сопоставленным с вершиной v_i , если в ней существует объект класса c с координатами относительно робота (p_x, p_y) , для которого выполняется условие сопоставления $matched(d, v_i)$:

$$matched(d, v_i) \Leftrightarrow \left\| \begin{pmatrix} p_x - o_x \\ p_y - o_y \end{pmatrix} \right\|_2 < r_{pos}, \quad (8)$$

где r_{pos} – порог допустимого отклонения координат. Семантическая оценка вычисляется как доля сопоставленных объектов:

$$s(v_i) = \frac{|\{d \in D_{cur} : matched(d, v_i)\}|}{|D_{cur}|}, \quad (9)$$

где D_{cur} – множество объектов, обнаруженных в текущем кадре. Если после переранжирования наилучшая вершина $v^* = \arg \max_{v \in V_{nav}} score(v)$ имеет два и более семантических объекта, сопоставленных с текущим кадром, то проводится дополнительное уточнение оценки положения робота относительно вершины v^* . По координатам объектов вершины v^* и координатам сопоставленных им объектов в текущем кадре вычисляется оптимальное положение робота относительно v^* с помощью метода наименьших квадратов:

$$\min_{x_m, y_m, \theta} \sum_{j=1}^k \left\| R^T(\theta) \begin{pmatrix} l_x^{(j)} - x_m \\ l_y^{(j)} - y_m \end{pmatrix} - \begin{pmatrix} o_x^{(j)} \\ o_y^{(j)} \end{pmatrix} \right\|, \quad (10)$$

где $(l^{(j)} \in \mathbb{R}^2, o^{(j)} \in \mathbb{R}^2)$ – множество сопоставленных пар двумерных координат объектов, $R^T(\theta) \in SO(2)$ – матрица поворота координат на угол θ .

4. Экспериментальное исследование

4.1. Постановка эксперимента

Эксперименты по оценке качества предложенного метода проводились в симуляционной среде, представляющей собой модель помещения большой площади. Эксперименты проводились в два этапа. Вначале выполнялся обход среды виртуальным агентом по заранее заданному маршруту, и во время обхода строился граф сцены с помощью предложенного метода. Затем выполнялась серия навигационных заданий с заданными точками старта и финиша, в ходе которых выполнялась локализация виртуального агента по построенному на первом этапе графу и оценивалась ошибка локализации. Навигация осуществлялась средствами симулятора по кратчайшему пути от старта до финиша. Ошибка локализации измерялась как среднее расхождение между истинным положением робота и вычисленным с помощью локализации на каждом шаге траектории.

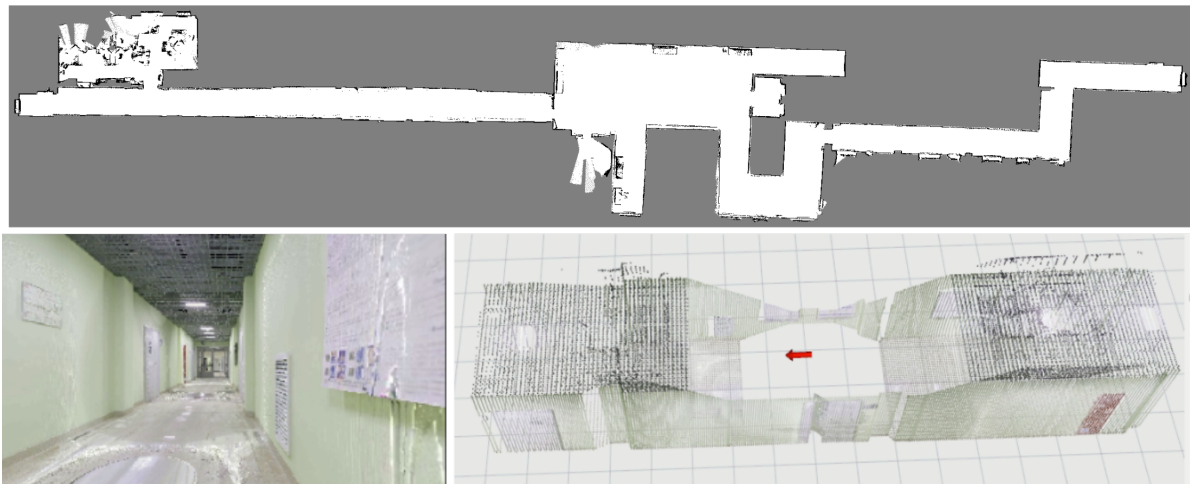


Рис. 4. Карта среды, используемой для экспериментов (вверху), изображение и облако точек из симулятора (внизу)

Для экспериментов использовался симулятор Habitat [10]. В качестве сцены использовалась модель пятого этажа корпуса «Цифра» Московского физико-технического института. Площадь сцены составила 1600 м², длина маршрута для обхода – 300 м. Пример сцены показан на рисунке 4. Ошибка локализации оценивалась как среднее отклонение истинной позиции робота от позиции, соответствующей состоянию робота в графе, определенному с помощью локализации. Движение и локализация выполнялись асинхронно – виртуальный агент двигался к целевой точке независимо от того, был ли завершен текущий шаг локализации.

4.2. Результаты

Результаты эксперимента представлены в таблице 1. Тестировалось четыре варианта локализации – без использования семантической информации и с семантическим переранжированием по всем трем вышеописанным формулам. Эксперименты проводились на компьютере со следующими характеристиками: 6-ядерный процессор Intel Core i5-11500, 32 ГБ оперативной памяти, видеокарта GeForce RTX 2060 с 6 ГБ видеопамати. Помимо ошибки локализации по построенному графу сцены, проводились измерения задержки локализации (времени от получения сенсорных данных до вычисления состояния робота в графе), а также размера построенного графа в хранилище.

Таблица 1. Результаты экспериментов с различным весом семантики и различными режимами переранжирования

α	Переранжирование	Ошибка локализации, м	Задержка локализации, с	Размер графа, МБ
0,0	–	2,10	0,80	0,374
0,3	Jaccard	1,20	0,87	0,403
0,3	IDF	1,36	0,87	0,403
0,3	Position	1,18	0,87	0,403

Как видно из таблицы, использование семантической информации снижает ошибку локализации на 35–44%, в зависимости от метода переранжирования. Наилучший результат показывает позиционно-осознанный режим переранжирования, сокращая ошибку локализации почти в два раза. Высокая ошибка локализации (более 2 м) без семантики обусловлена наличием в сцене длинных однообразных коридоров, разные участки которых имеют одинаковую геометрическую форму при проекции на плоскость. Добавление семантической информации позволяет различать участки коридоров по дверям, мебели, пожарным шкафам и т. д., что уменьшает число ложных локализаций и, как следствие, среднюю ошибку. При добавлении семантической информации задержка локализации и потребление памяти построенным графом увеличиваются незначительно. Граф сцены помещения площадью 1600 м² занимает всего 400 КБ дискового пространства – на порядки меньше, чем графы сцены, построенные традиционными методами, и метрические карты.

Заключение

В данной работе предложен метод построения компактного графа сцены, пригодный для автономной навигации мобильных роботов в помещениях большой площади. Метод строит граф, состоящий из двух слоев: навигационного подграфа, описывающего пространственную связность локаций, и семантического слоя, содержащего информацию об объектах, обнаруженных в локациях. Построение навигационного подграфа и локализация в нем выполняются методом PRISM-ТороМар, в котором локализация проводится в два этапа: поиск текущей локации методом распознавания мест и уточнение позиции методом сопоставления двумерных проекций сканов. Объекты семантического слоя используются для переупорядочивания локаций-кандидатов

навигационного графа, найденных с помощью распознавания мест. Предложено три режима переупорядочивания: на основе коэффициента Жаккара, IDF-взвешенного коэффициента и позиционно-осознанного сопоставления объектов. В последнем режиме также проводится уточнение положения робота с использованием позиций объектов.

Эксперименты, проведенные в симуляционной среде большой площади, показали, что граф сцены, построенный предложенным методом, занимает незначительно больше памяти, чем компактный топологический граф метода PRISM-ТороМар без семантики (400 КБ на сцену площадью 1600 м²), при этом ошибка локализации по графу с добавлением семантической информации уменьшается почти в два раза. В дальнейшем планируется доработка предложенного метода и экспериментальное сравнение с другими методами построения графов сцены, настройка и обучение весового коэффициента α , а также проведение экспериментов с навигацией по построенному графу сцены и переход из симуляции на реальные робототехнические платформы.

Список литературы

- [1] Labbé, M. RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation / M. Labbé, F. Michaud // *Journal of Field Robotics*. – 2019. – Vol. 36, No. 2. – P. 416–446. – DOI 10.1002/rob.21831.
- [2] GLIM: 3D range-inertial localization and mapping with GPU-accelerated scan matching factors / K. Koide, M. Yokozuka, S. Oishi, A. Banno // *Robotics and Autonomous Systems*. – 2024. – Vol. 179. – P. 104750. – DOI 10.1016/j.robot.2024.104750.
- [3] Muravyev, K. Evaluation of RGB-D SLAM in large indoor environments / K. Muravyev, K. Yakovlev // *International Conference on Interactive Collaborative Robotics*. – 2022. – P. 93–104. – DOI 10.1007/978-3-031-23609-9_9.
- [4] Muravyev, K. Evaluation of topological mapping methods in indoor environments / K. Muravyev, K. S. Yakovlev // *IEEE Access*. – 2023. – Vol. 11. – P. 132683–132698. – DOI 10.1109/ACCESS.2023.3335818.
- [5] PRISM-TopoMap: Online topological mapping with place recognition and scan matching / K. Muravyev, A. Melekhin, D. Yudin, K. Yakovlev // *IEEE Robotics and Automation Letters*. – 2025. – Vol. 10, No. 4. – P. 3126–3133. – DOI 10.1109/LRA.2025.3541454.
- [6] Topological semantic graph memory for image-goal navigation / N. Kim, O. Kwon, H. Yoo [et al.] // *Proceedings of the 6th Conference on Robot Learning (CoRL)*, PMLR. – 2023. – Vol. 205. – P. 393–402.
- [7] Clio: Real-time task-driven open-set 3D scene graphs / D. Maggio, Y. Chang, N. Hughes [et al.] // *IEEE Robotics and Automation Letters*. – 2024. – Vol. 9, No. 10. – P. 8921–8928. – DOI 10.1109/LRA.2024.3451395.
- [8] Hughes, N. Hydra: A real-time spatial perception system for 3D scene graph construction and optimization / N. Hughes, Y. Chang, L. Carlone // *arXiv*. – 2022. – arXiv:2201.13360. – DOI 10.48550/arXiv.2201.13360.
- [9] Khronos: Middleware for simplified time management in CPS / S. Peros, S. Delbruel, S. Michiels [et al.] // *Proceedings of the 13th ACM International Conference on Distributed and Event-based Systems*. – 2019. – P. 127–138. – DOI 10.1145/3328905.3329507.
- [10] Habitat: A platform for embodied AI research / M. Savva, A. Kadian, O. Maksymets [et al.] // *Proceedings of the IEEE/CVF International Conference on Computer Vision*. – 2019. – P. 9338–9346. – DOI 10.1109/ICCV.2019.00943.
- [11] A modular robotic system for autonomous exploration and semantic updating in large-scale indoor environments / S. H. Allu, I. Kadosh, T. Summers, Y. Xiang // *arXiv*. – 2024. – arXiv:2409.15493. – DOI 10.48550/arXiv.2409.15493.

- [12] Carpenter, G. A. Adaptive resonance theory : Tech. Rep. / G. A. Carpenter, S. Grossberg. – Boston, MA, USA : Boston University, Center for Adaptive Systems and Department of Cognitive and Neural Systems, 1993.
- [13] TopoNav: Topological navigation for efficient exploration in sparse reward environments / J. Hossain, A. Z. M. Faridee, N. Roy [et al.] // 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). – 2024. – P. 693–700. – DOI 10.1109/IROS58592.2024.10802380.
- [14] SemanticTopoLoop: Semantic loop closure with 3D topological graph based on quadric-level object map / Z. Cao, Q. Zhang, J. Guang [et al.] // IEEE Robotics and Automation Letters. – 2024. – Vol. 9, No. 5. – P. 4257–4264. – DOI 10.1109/LRA.2024.3374169.
- [15] Learning 3D semantic scene graphs from 3D indoor reconstructions / J. Wald, H. Dharmo, N. Navab, F. Tombari // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2020. – P. 3961–3970. – DOI 10.1109/CVPR42600.2020.00402.
- [16] Exploiting edge-oriented reasoning for 3D point-based scene graph analysis / C. Zhang, J. Yu, Y. Song, W. Cai // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2021. – P. 9705–9715. – DOI 10.1109/CVPR46437.2021.00958.
- [17] SGFormer++: Semantic graph transformer for incremental 3D scene graph generation / M. Qi, C. Lv, Z. Fu [et al.] // arXiv. – 2026. – arXiv:2606.15328. – DOI 10.48550/arXiv.2606.15328.
- [18] Learning transferable visual models from natural language supervision / A. Radford, J. W. Kim, C. Hallacy [et al.] // Proceedings of the 38th International Conference on Machine Learning, PMLR. – 2021. – Vol. 139. – P. 8748–8763.
- [19] SGR3D: Self-supervised 3D scene graph learning via object-level scene reconstruction / S. Koch, N. Vaskevicius, M. Colosi [et al.] // Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. – 2024. – P. 3392–3402. – DOI 10.1109/WACV57701.2024.00337.
- [20] vS-Graphs: Integrating visual SLAM and situational graphs through multi-level scene understanding / A. Tourani, S. Ejaz, H. Bavle [et al.] // arXiv. – 2025. – arXiv:2503.01783. – DOI 10.48550/arXiv.2503.01783.
- [21] S-Graphs+: Real-time localization and mapping leveraging hierarchical representations / H. Bavle, J. L. Sánchez-López, M. Shaheer [et al.] // IEEE Robotics and Automation Letters. – 2023. – Vol. 8, No. 8. – P. 4927–4934. – DOI 10.1109/LRA.2023.3290512.
- [22] ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning / Q. Gu, A. Kuwajerwala, S. Morin [et al.] // 2024 IEEE International Conference on Robotics and Automation (ICRA). – 2024. – P. 5021–5028. – DOI 10.1109/ICRA57147.2024.10610243.
- [23] Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection / S. Liu, Z. Zeng, T. Ren [et al.] // Computer Vision – ECCV 2024, Springer. – 2024. – Vol. 47. – P. 38–55. – DOI 10.1007/978-3-031-72970-6_3.

BUILDING COMPACT SCENE GRAPHS BASED ON A TOPOLOGICAL MAP FOR AUTONOMOUS NAVIGATION OF A MOBILE ROBOT

Muravyev K.F.^{1, a}, Romanenko V.I.^{2, b}

¹ Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences (FRC CSC RAS), Moscow, Russia.

² National Research University Higher School of Economics (NRU HSE), Moscow, Russia.

E-mail: ^a kmuravev@frccsc.ru, ^b viktorsamr@yandex.ru

Abstract. Autonomous navigation of a mobile robot in human-centered environments requires a map that contains not only a geometric model of the environment for path planning, but also information about environmental objects (doors, furniture, office equipment, etc.). Scene graphs provide such a map representation, where nodes correspond to rooms, locations, and objects, and edges encode spatial connectivity or relationships between objects. Most modern scene graph construction methods have high computational complexity, and the graphs they produce are redundant for the purposes of autonomous robot navigation. This paper proposes a method for constructing Compact Scene Graphs (CSG), which is based on the computationally efficient topological mapping method PRISM-TopoMap and the association of semantic objects with locations on the topological map. The resulting scene graph enables route planning to objects via topological map locations and achieves high-precision localization. The proposed method was experimentally evaluated in the Habitat simulation environment. The experimental results demonstrate that the proposed CSG consumes significantly less memory than traditional metric maps and scene graphs, while providing reliable localization on the topological map through association with semantic objects.

Keywords: *scene graph, topological map, semantic segmentation, autonomous navigation, simulated environment.*

References

- [1] Labbé, M., & Michaud, F. (2019). 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*, 36(2), 416–446. <https://doi.org/10.1002/rob.21831>
- [2] Koide, K., Yokozuka, M., Oishi, S., & Banno, A. (2024). GLIM: 3D range-inertial localization and mapping with GPU-accelerated scan matching factors. *Robotics and Autonomous Systems*, 179, Article 104750. <https://doi.org/10.1016/j.robot.2024.104750>
- [3] Muravyev, K., & Yakovlev, K. (2022). Evaluation of RGB-D SLAM in large indoor environments. In *International Conference on Interactive Collaborative Robotics* (pp. 93–104). https://doi.org/10.1007/978-3-031-23609-9_9

For citation:

Muravyev K.F., Romanenko V.I. Building Compact Scene Graphs Based on a Topological Map for Autonomous Navigation of a Mobile Robot. *Intelligent transport*, 2026, no. 3(39), pp. 37–52. DOI: 10.24412/3033-6007-2026-339-37-52. (in Russian)

- [4] Muravyev, K., & Yakovlev, K. S. (2023). Evaluation of topological mapping methods in indoor environments. *IEEE Access*, 11, 132683–132698. <https://doi.org/10.1109/ACCESS.2023.3335818>
- [5] Muravyev, K., Melekhin, A., Yudin, D., & Yakovlev, K. (2025). PRISM-TopoMap: Online topological mapping with place recognition and scan matching. *IEEE Robotics and Automation Letters*, 10(4), 3126–3133. <https://doi.org/10.1109/LRA.2025.3541454>
- [6] Kim, N., Kwon, O., Yoo, H., Choi, Y., Park, J., & Oh, S. (2023). Topological semantic graph memory for image-goal navigation. In *Proceedings of the 6th Conference on Robot Learning (CoRL)* (Vol. 205, pp. 393–402). PMLR.
- [7] Maggio, D., Chang, Y., Hughes, N., Trang, M., Griffith, D., Dougherty, C., Cristofalo, E., Schmid, L., & Carlone, L. (2024). Clio: Real-time task-driven open-set 3D scene graphs. *IEEE Robotics and Automation Letters*, 9(10), 8921–8928. <https://doi.org/10.1109/LRA.2024.3451395>
- [8] Hughes, N., Chang, Y., & Carlone, L. (2022). Hydra: A real-time spatial perception system for 3D scene graph construction and optimization (arXiv:2201.13360). arXiv. <https://doi.org/10.48550/arXiv.2201.13360>
- [9] Peros, S., Delbruel, S., Michiels, S., Joosen, W., & Hughes, D. (2019). Khronos: Middleware for simplified time management in CPS. In *Proceedings of the 13th ACM International Conference on Distributed and Event-based Systems* (pp. 127–138). <https://doi.org/10.1145/3328905.3329507>
- [10] Savva, M., Kadian, A., Maksymets, O., Zhao, Y., Wijmans, E., Jain, B., Straub, J., Liu, J., Koltun, V., Malik, J., Batra, D., & Mottaghi, R. (2019). Habitat: A platform for embodied AI research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 9338–9346). <https://doi.org/10.1109/ICCV.2019.00943>
- [11] Allu, S. H., Kadosh, I., Summers, T., & Xiang, Y. (2024). A modular robotic system for autonomous exploration and semantic updating in large-scale indoor environments (arXiv:2409.15493). arXiv. <https://doi.org/10.48550/arXiv.2409.15493>
- [12] Carpenter, G. A., & Grossberg, S. (1993). *Adaptive resonance theory* (Technical Report). Boston University, Center for Adaptive Systems and Department of Cognitive and Neural Systems.
- [13] Hossain, J., Faridee, A. Z. M., Roy, N., Freeman, J., Gregory, T., & Trout, T. (2024). TopoNav: Topological navigation for efficient exploration in sparse reward environments. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 693–700). <https://doi.org/10.1109/IROS58592.2024.10802380>
- [14] Cao, Z., Zhang, Q., Guang, J., Wu, S., Hu, Z., & Liu, J. (2024). SemanticTopoLoop: Semantic loop closure with 3D topological graph based on quadric-level object map. *IEEE Robotics and Automation Letters*, 9(5), 4257–4264. <https://doi.org/10.1109/LRA.2024.3374169>
- [15] Wald, J., Dharmo, H., Navab, N., & Tombari, F. (2020). Learning 3D semantic scene graphs from 3D indoor reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 3961–3970). <https://doi.org/10.1109/CVPR42600.2020.00402>
- [16] Zhang, C., Yu, J., Song, Y., & Cai, W. (2021). Exploiting edge-oriented reasoning for 3D point-based scene graph analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9705–9715). <https://doi.org/10.1109/CVPR46437.2021.00958>
- [17] Qi, M., Lv, C., Fu, Z., Zhang, X., & Ma, H. (2026). SGFormer++: Semantic graph transformer for incremental 3D scene graph generation (arXiv:2606.15328). arXiv. <https://doi.org/10.48550/arXiv.2606.15328>
- [18] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning* (Vol. 139, pp. 8748–8763). PMLR.
- [19] Koch, S., Vaskevicius, N., Colosi, M., Hermosilla, P., & Ropinski, T. (2024). SGRec3D: Self-supervised 3D scene graph learning via object-level scene reconstruction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 3392–3402). <https://doi.org/10.1109/WACV57701.2024.00337>

- [20] Tourani, A., Ejaz, S., Bavle, H., Morilla-Cabello, D., Sanchez-Lopez, J. L., & Voos, H. (2025). vS-Graphs: Integrating visual SLAM and situational graphs through multi-level scene understanding (arXiv:2503.01783). arXiv. <https://doi.org/10.48550/arXiv.2503.01783>
- [21] Bavle, H., Sánchez-López, J. L., Shaheer, M., Civera, J., & Voos, H. (2023). S-Graphs+: Real-time localization and mapping leveraging hierarchical representations. *IEEE Robotics and Automation Letters*, 8(8), 4927–4934. <https://doi.org/10.1109/LRA.2023.3290512>
- [22] Gu, Q., Kuwajerwala, A., Morin, S., Jatavallabhula, K. M., Sen, B., Agarwal, A., Rivera, C., Paul, W., Ellis, K., Chellappa, R., Gan, C., de Melo, C. M., Tenenbaum, J. B., Torralba, A., Shkurti, F., & Paull, L. (2024). ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 5021–5028). <https://doi.org/10.1109/ICRA57147.2024.10610243>
- [23] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., Zhu, J., & Zhang, L. (2024). Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. In *Computer Vision – ECCV 2024* (Vol. 47, pp. 38–55). Springer. https://doi.org/10.1007/978-3-031-72970-6_3