

# Топологическое картирование помещений с использованием нейросетевой локализации и сопоставления сканов

К. Ф. Муравьев

Федеральный исследовательский центр «Информатика и управление» РАН, Москва, Россия

**Аннотация.** В статье представлен метод топологического картирования помещений, способный строить и обновлять граф локаций по облакам точек и одометрии без использования глобальных метрических координат. В данном методе для локализации в графе используется нейросетевое распознавание локации совместно с сопоставлением сканов по двумерным проекциям. Было проведено экспериментальное исследование предложенного подхода на нескольких фотореалистичных симуляционных средах, а также на данных с реального робота. В симуляции также проводится сравнение с лучшими современными методами топологического картирования. Результаты сравнения подтверждают, что предложенный метод значительно опережает остальные по навигационной эффективности, сохраняя связность графа, высокую площадь покрытия сцены и низкую долю неконсистентных ребер.

**Ключевые слова:** одновременное картирование и локализация (SLAM), топологическая карта, мобильные роботы.

DOI 10.14357/20718632240303

EDN CFSQSG

## Введение

Традиционные методы картирования строят плотную метрическую карту (например, в виде сетки занятости). Поддержание такой карты при долговременной навигации затруднительно ввиду высоких затрат вычислительных ресурсов и накопления ошибки одометрии. Представление пространства в виде разреженной топологической структуры (например, графа локаций) позволяет устранить эти недостатки и обеспечить быстрое планирование пути.

Построение точной карты является одной из ключевых задач автономной навигации мобильных роботов, и является критически важным для навигации мобильных роботов [1]. Некоторые методы картирования, описанные в работе [2],

строят разреженную трехмерную карту, на которую наносятся только координаты особых точек. Другие общеизвестные методы картирования, такие как RTAB-MAP [3] или Cartographer [4], строят карту в виде плотной двумерной или трехмерной метрической структуры, такой как сетка занятости или облако точек. Однако такие плотные метрические карты требуют больших объемов памяти для хранения, что может привести к переполнению памяти бортового компьютера робота при навигации в среде большой площади [5]. Также метрические карты подвержены накоплению ошибки одометрии, что может привести к сбоям в картировании и замыкании циклов с ростом размера карты.

Альтернативным подходом к построению карты является топологическое картирование –

представление карты в виде графовой структуры (например, графа коридоров [6] или графа выпуклых кластеров свободного пространства [7]). При использовании топологических свойств среды (связность, относительное расположение локаций) значительно снижается потребление памяти и вычислительные затраты на обновление карты. Также, в методах, не использующих глобальные метрические координаты, исключается накопление ошибки позиционирования [8; 9]. Кроме того, благодаря разреженности топологических карт планирование пути в них происходит значительно быстрее, чем в метрических картах [7].

Методы топологического картирования разделяются на несколько категорий. Так, довольно большой класс методов (например, [7; 10]) строит топологическую карту по предварительно построенной метрической карте. Еще одна группа методов, таких как [11, 12], строит топологическую карту в паре с метрической картой по сенсорным данным. Такие методы дают полное представление среды и обеспечивают быстрое планирование пути. Однако они являются довольно ресурсозатратными, а также подвержены накоплению ошибки в силу наличия глобальной метрической карты.

В последние годы также появилось множество обучаемых методов построения топологической карты, работающих без использования глобальных метрических координат. Большинство таких методов (например, [13; 14]) разработаны для решения конкретных задач, таких как навигация до целевого объекта в симуляционной среде, и не опробованы на данных с реального робота. Также, подобные методы, как правило, полагаются исключительно на предсказанные нейросетью дескрипторы, что может привести к добавлению в граф ребер между далекими друг от друга локациями и сбоям в навигации. Некоторые другие методы, такие как [15], функционируют в реальных средах и используют алгоритмы фильтрации ребер, однако требуют предварительно построенной топологической карты на старте.

Одной из наиболее распространенных топологических структур для навигации является граф локаций (таких как комната, коридор, лифтовый холл и т.д.), в котором ребра соединяют

смежные локации. Такой граф обеспечивает быстрое и удобное планирование пути, однако локализация в графе (привязка робота к локации) является нетривиальной задачей. Для решения этой задачи обычно проводится сравнение текущего наблюдения робота с локациями графа по предсказанным нейросетевыми методами дескрипторам [16]. Однако подобные методы могут сопоставить несмежные локации из-за их визуального сходства и близости предсказанных дескрипторов, что может спровоцировать сбой навигации. Например, эксперименты проведенные в работе [17], показывают, что современный нейросетевой метод топологического картирования TSGM [13] соединяет ребрами локации, находящиеся в противоположных концах здания.

В данной работе представлен метод построения графа локаций, использующий сопоставление текущего наблюдения робота с локациями графа по нейросетевым дескрипторам, а также удаление ложных сопоставлений и определение относительных позиций от робота до точек наблюдения локаций путем классического сопоставления сканов. Представленный метод принимает на вход облака точек с бортовых сенсоров восприятия (RGB-D камер или лидаров), а также локальную одометрию. Выходом является граф локаций, покрывающий всю пройденную роботом часть среды. Проводится экспериментальное исследование представленного метода в сравнении с тремя современными методами топологического картирования: Hydra [11], IncrementalTopo [12] и TSGM [13].

## 1. Постановка задачи

Рассмотрим робототехническую систему, оснащенную сенсором восприятия (RGB-D камера или лидар) и сенсором одометрии, движущуюся внутри помещений по предварительно заданной траектории. Задачей робота является построение и непрерывное обновление графа локаций, который может быть использован для дальнейшего планирования миссии и навигации.

Окружающая среда представляется в виде пространства  $W \subset \mathbb{R}^2$ , которое делится на свободное пространство и множество препятствий:  $W = W_{free} \cup W_{obs}$  ( $W_{free} \cap W_{obs} = \emptyset$ ). Лока-

ция  $loc \subset W$  представляет собой участок пространства среды, например, комнату или лифтовый холл. Для каждой локации задается точка наблюдения  $loc_{obs}$ , а также карта признаков  $F(loc) = FMap(loc_{obs})$ , которая строится по наблюдению из точки  $loc_{obs}$ . Такая карта признаков может быть, к примеру, дескриптором, предсказанным по входному облаку точек с помощью нейронной сети.

Две локации  $loc$  и  $loc'$  называются смежными, если их пересечение содержит точки свободного пространства среды:  $loc \cap loc' \cap W_{free} \neq \emptyset$ . Смежные локации соединяются ребром в графе локаций:  $e = (loc, loc')$ . Каждое ребро ассоциировано с картой признаков ребра  $F(e)$ , при помощи которой робот будет двигаться между локациями. Такая карта признаков может быть направлением от первой локации ребра до второй, или частью панорамного изображения с первой локации, на которой наблюдается вторая локация, или любой другой характеристикой, способствующей навигации между локациями.

Для обеспечения успешного планирования путей и эффективной навигации граф локаций должен быть связным и покрывающим всю площадь, исследованную роботом. Более того, для безопасной и эффективной навигации, ребра графа должны соединять только локации, являющиеся смежными в среде, при этом оптимальные пути по ребрам графа должны быть по длине близки к оптимальным путям в среде. Исходя из вышеуказанных требований, качество топологического картирования оценивается с использованием следующих метрик:

- Связность: число компонент связности в графе.

- Площадь, покрытая главной компонентой:  $Coverage = \frac{Area(\cup_{loc \in V_{main}} loc)}{Area(W)}$ ,

где  $(V_{main}, E_{main})$  – это главная компонента связности графа локаций.

- Доля неконсистентных ребер (ребер, соединяющих не смежные в среде локации):

$$PIE = \frac{|\{(u,v) \in E: u \cap v \cap W_{free} = \emptyset\}|}{|E|}.$$

- Эффективность навигации в графе, вычисляемая как значение SPL (Success weighted by Path Length – успешность, взвешенная по длине пути), усредненное по  $N$  случайно выбранным парам вершин  $(u_i, v_i), i = 1, \dots, N$ :

$$SPL = \sum_{i=1}^N \frac{|Path(u_i, v_i, W_{free})| \cdot I_{cons}(Path(u_i, v_i, G))}{|Path(u_i, v_i, G)|},$$

где  $Path(u, v, G)$  обозначает кратчайший путь между вершинами  $u$  и  $v$  в графе  $G$ ;  $Path(u, v, W_{free})$  – кратчайший путь в среде между точками наблюдения локаций  $u$  и  $v$ ;  $I_{cons}(Path(u, v, G))$  равно 1 в случае, если путь  $Path(u, v, G)$  существует и не содержит неконсистентных ребер, и 0 в противном случае.

## 2. Предложенный метод топологического картирования

Предложенный в работе метод строит и обновляет граф локаций окружающей среды в реальном времени по облакам точек и одометрии с сенсоров робота, а также по результатам локализации. За основу предложенного метода взят метод PRISM-ТороМар, описанный в работе [18]. Каждая локация в графе задается облаком точек, снятым с робота в точке наблюдения этой локации, а также нейросетевым дескриптором для локализации. Также, на каждом шаге  $t$  работы метода поддерживается локация  $v_{cur}^t$  и относительная позиция от ее точки наблюдения до робота  $T_{cur}^t$ . Входом метода на шаге  $t$  являются граф локаций  $G_{t-1} = (V_{t-1}, E_{t-1})$  с текущим состоянием робота  $(v_{cur}^{t-1}, T_{cur}^{t-1})$ , результаты локализации  $\{(v_{loc}, T_{loc})\}_{loc \in Loc}$  – локализованные локации и позиции робота относительно этих локаций, одометрия  $o_t$  и облако точек  $C_t$ , снятое с позиции робота. Выходом является обновленный граф локаций  $G_t = (V_t, E_t)$  и обновленное состояние робота в графе:  $v_{cur}^t$  и  $T_{cur}^t$ . Схема метода изображена на Рис. 1.

Процесс обновления графа состоит из следующих шагов:

- 1) Проверка того, что робот находится в пределах локации  $v_{cur}^{t-1}$ , и текущий скан с него перекрывается с локацией  $v_{cur}^{t-1}$  по достаточному проценту площади. Если проверка пройдена, то  $T_{cur}$  обновляется по одометрии:

$$v_{cur}^t = v_{cur}^{t-1}, T_{cur}^t = T_{cur}^{t-1} \cdot o_t.$$

- 2) Если робот находится за пределами  $v_{cur}^{t-1}$ , или площадь перекрытия сканов мала, то сначала производится попытка обновить значение  $v_{cur}$  на одну из соседних с  $v_{cur}^{t-1}$  локаций в графе (т.е. перейти по ребру). Для этого сканы локаций

$v_{cur}^{t-1}$  и соседей сопоставляются с помощью детектора углов Харриса [19]. Относительные позиции, записанные на ребрах, используются в качестве начальной догадки для сопоставления. Если среди соседей  $v_{cur}^{t-1}$  найдена подходящая для перехода по ребру локация  $v_{next}$  (т.е. сканы успешно сопоставлены, робот находится в пределах локации  $v_{next}$  и расстояние от позиции робота до точки наблюдения  $v_{next}$  меньше, чем до точки наблюдения  $v_{cur}^{t-1}$ ), то производится переход по ребру из  $v_{cur}^{t-1}$  в  $v_{next}$ :

$$v_{cur}^t = v_{next}; T_{cur}^t = T_{next},$$

где  $T_{next}$  – найденная путем сопоставления сканов позиция робота относительно точки наблюдения  $v_{next}$ .

3) В противном случае проводится попытка сменить значение  $v_{cur}$  на одну из локализованных локаций. Для этого сканы локализованных

локаций сдвигаются на относительную позицию, предсказанную локализацией, после чего вычисляется доля перекрытия этих сканов с текущим сканом робота. Если найдена локация  $v_{loc}$  с достаточной долей перекрытия, она становится текущей локацией и соединяется ребром с локацией  $v_{cur}^{t-1}$ :

$$v_{cur}^t = v_{loc}; T_{cur}^t = T_{loc};$$

$$E_t = E_{t-1} \cup \{(v_{cur}^{t-1}, v_{loc})\}.$$

4) Если подходящая локация не нашлась среди локализованных, в граф добавляется новая локация  $v_{new}$ , наблюдаемая с текущей позиции робота. После добавления локация  $v_{new}$  становится новой текущей локацией и соединяется ребрами со всеми локациями из результатов локализации:

$$v_{cur}^t = v_{new}; T_{cur}^t = I;$$

$$E_t = E_{t-1} \cup \{(v_{new}, v_{cur}^{t-1})\} \cup \{(v_{new}, v_{loc})_{loc \in Loc}\}.$$

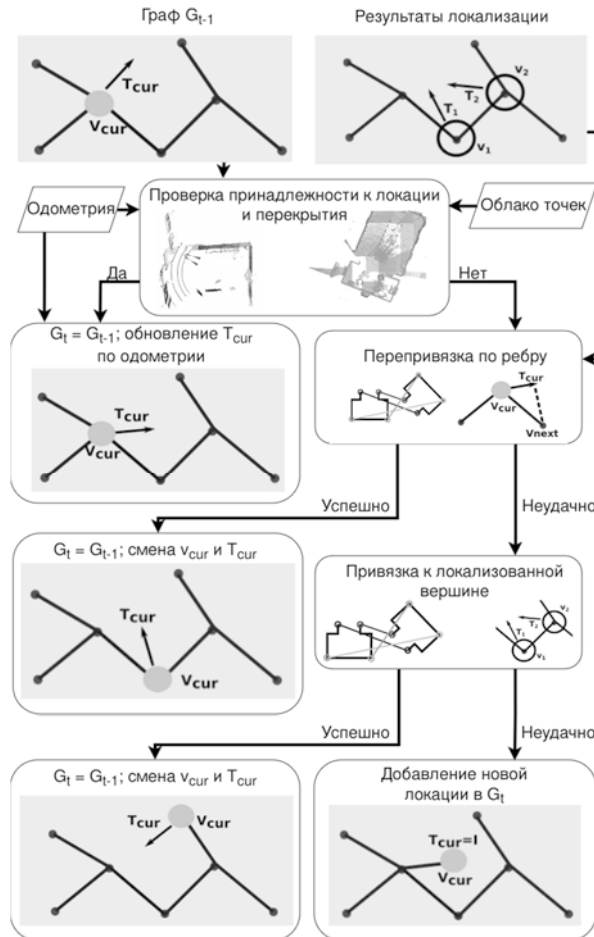


Рис. 1. Схема обновления графа и поддержания текущего состояния в нем: проверка нахождения робота внутри локации  $v_{cur}^{t-1}$ , смена  $v_{cur}$  по ребрам и результатам локализации, добавление новой локации

Также для более эффективной навигации производится поиск и замыкание циклов, проходящих через текущую позицию робота. Для поиска проверяется каждая пара локализованных вершин. Если путь между этими двумя вершинами в графе  $G_{t-1}$  значительно длиннее, чем путь между ними через текущую позицию робота, то цикл замыкается созданием новой локации, наблюдаемой с текущей позиции робота, и соединением ее ребрами со всеми локализованными вершинами. Во всех случаях добавления новой вершины она соединяется ребром с предыдущей локацией, что гарантирует связность построенного графа.

### 3. Локализация в графе с помощью распознавания мест и сопоставления сканов

Схема алгоритма локализации в графе локаций представлена на Рис. 2. На вход подается облако точек  $C_t$ , снятое сенсором робота в момент  $t$ , а также текущий граф  $G_{t-1}$ . Сначала проводится вокселизация (понижение дискретизации) облака точек, и по вокселизованному облаку нейросетевым методом  $F_{PR}$  предсказывается дескриптор  $d$ . Предсказанный дескриптор сравнивается с дескрипторами облаков точек всех локаций графа, и для рассмотрения выбирается  $k$  наиболее похожих на текущую локаций, (т.е. таких локаций, дескрипторы которых наиболее близки к  $d$ ). Затем для этих  $k$  локаций ищутся позиции робота относительно их точек наблюдения. Для поиска относительных позиций используется алгоритм сопоставления сканов,

описанный ниже и основанный на двумерных признаках, извлекаемых из проекции облака точек. Локации, чьи сканы не сопоставились со сканом с робота, удаляются из рассмотрения. Оставшиеся локации считаются локализованными и вместе с относительными позициями подаются на вход метода обновления графа.

Фильтрация похожих локаций путем сопоставления сканов позволяет устранить ошибки нейросетевой локализации и избежать добавления неконсистентных ребер в граф. Также получаемая в ходе сопоставления оценка относительной позиции позволяет скорректировать зашумленную оценку позиции по одометрии и задавать более точные относительные позиции ребрам графа локаций для повышения навигационной эффективности.

Алгоритм сопоставления сканов принимает на вход пару облаков точек и вычисляет преобразование координат между точками наблюдения первого и второго облака, либо (в случае ненахождения преобразования) выдает сообщение об ошибке. В начале работы алгоритма оба облака точек проецируются на плоскость, переводятся в формат воксельной сетки и затем проецируются в черно-белое изображение. Затем из полученных изображений извлекаются особые точки детектором ORB [20] и сопоставляются с помощью метода FLANN [21]. Пример сопоставления особых точек показан на Рис. 3. Далее искомое преобразование координат вычисляется методом наименьших квадратов после  $N$  итераций удаления выбросов. На каждой итерации  $i$  по текущему набору сопоставлений вычисляется преобразование координат методом

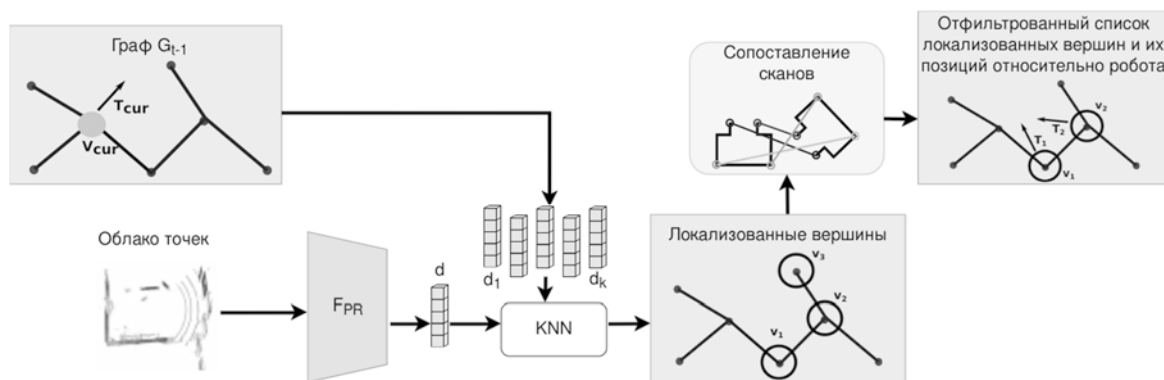


Рис. 2. Локализация в графе локаций: распознавание места по дескрипторам, предсказанным нейросетевым энкодером  $F_{PR}$ , фильтрация и нахождение относительных позиций путем сопоставления сканов

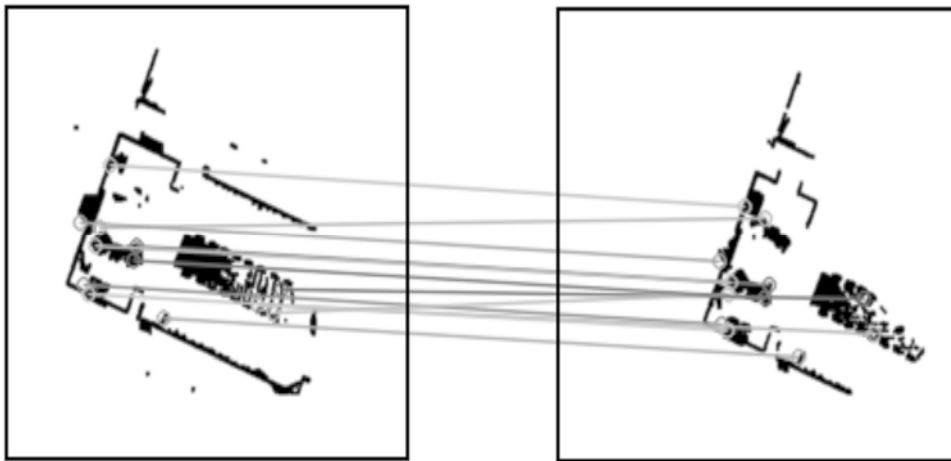


Рис. 3. Пример сопоставления облаков точек путем извлечения особых точек из двумерных проекций

наименьших квадратов. Далее преобразование применяется к особым точкам первого облака, после чего преобразованные особые точки сравниваются с сопоставленными особыми точками второго облака. Если расстояние между ними превышает значение  $\delta_i$ , то данная пара удаляется из набора сопоставлений. Если сопоставлений осталось менее  $K$ , то сканы считаются не сопоставленными, и алгоритм выдает сообщение об ошибке. Если все  $N$  итераций успешно пройдены, то алгоритм выдает преобразование координат, полученное методом наименьших квадратов по оставшимся сопоставлениям.

В экспериментах в качестве  $F_{PR}$  использовался нейросетевой метод MinkLoc3D, описанный в работе [22]. Дообучение нейросети не

проводилось, использовались параметры, предоставленные авторами метода MinkLoc3D. Архитектура метода изображена на Рис. 4. На вход метод принимал облако точек, а на выходе давал дескриптор в виде вектора размерности 256. Для оценки близости дескрипторов использовалось евклидово расстояние в 256-мерном векторном пространстве. Значение  $k$  числа рассматриваемых ближайших локаций в графе было выбрано равным 5. Для быстрого поиска близких дескрипторов использовалась программная библиотека faiss<sup>1</sup>. Для алгоритма сопоставления сканов использовались следующие значения параметров:

$$N = 5; \delta_i = [2.5, 1, 0.5, 0.25, 0.25]; K = 5.$$

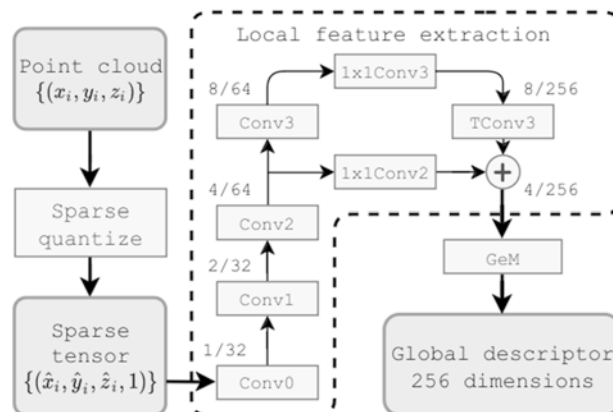


Рис. 4. Архитектура нейросети MinkLoc3D

<sup>1</sup> <https://github.com/facebookresearch/faiss>

#### 4. Экспериментальное исследование

Для проверки эффективности предложенного метода было проведено экспериментальное исследование на пяти симуляционных сценах из набора данных Matterport3D [23]. На каждой из сцен виртуальный агент, оснащенный панорамной RGB-D камерой, двигался по предварительно заданному маршруту. Площадь сцен варьировалась от 175 до 450 м<sup>2</sup>, длины траекторий – от 127 до 322 м. Примеры сцен, траекторий и панорамных RGB-D изображений представлены на Рис. 5.

Задачей агента было построение графа локаций, покрывающего всю сцену. Построенный граф оценивался по метрикам, описанным в разделе 1: количество компонент связности, площадь, покрытая главной компонентой, доля неконсистентных ребер, навигационная эффективность (SPL). С помощью этих метрик было проведено сравнение предложенного метода с двумя метрико-топологическими методами картирования (Hydra [11] и S-graphs+ [24]) и современным нейросетевым чисто топологическим

методом TSGM [13]. Результаты сравнения представлены в Табл. 1. Примеры графов, построенных методами, представлены на Рис. 6.

Как показано в таблице, представленный метод значительно превосходит по навигационной эффективности как классические метрико-топологические методы S-graphs+ и Hydra, так и нейросетевой чисто топологический метод TSGM. Значения SPL у методов Hydra и S-graphs+ составили менее 0.4, что обусловлено несвязностью построенных графов и низкой площадью покрытия сцены. Метод TSGM построил связные графы, но они имели высокую долю неконсистентных ребер из-за отсутствия методов фильтрации нейросетевого распознавания мест, что также привело к сравнительно низкому значению SPL (0.6). Предложенный метод построил связные графы с низкой долей неконсистентных ребер, что позволило достичь значения SPL, равного 0.77.

Помимо экспериментального исследования в симуляционных сценах, предложенный метод также был протестирован на данных с реального робота. Для тестирования использовался робот Clearpath Husky, оснащенный трехмерным

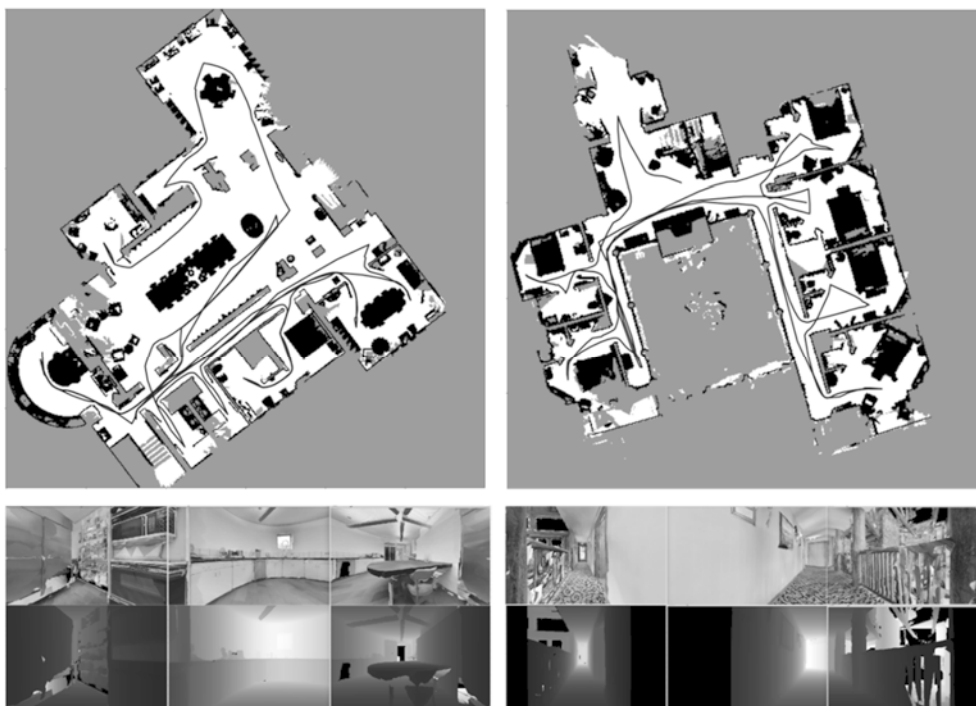


Рис. 5. Карта препятствий (показана вверх, черные пятна обозначают препятствия, белые – свободные зоны), траектория (показана сверху тонкой линией), панорамные изображения и глубины (показаны внизу) для двух сцен из экспериментального исследования

Табл. 1. Результаты экспериментов на симуляционных сценах

| Метод     | Связность  | Покрытие    | Доля неконсистентных ребер | SPL         |
|-----------|------------|-------------|----------------------------|-------------|
| Hydra     | 7.2        | 0.74        | <b>0.02</b>                | 0.39        |
| S-graphs+ | 6.2        | 0.54        | <b>0.02</b>                | 0.29        |
| TSGM      | <b>1.0</b> | <b>0.95</b> | 0.07                       | 0.60        |
| Ours      | <b>1.0</b> | 0.92        | <b>0.02</b>                | <b>0.77</b> |

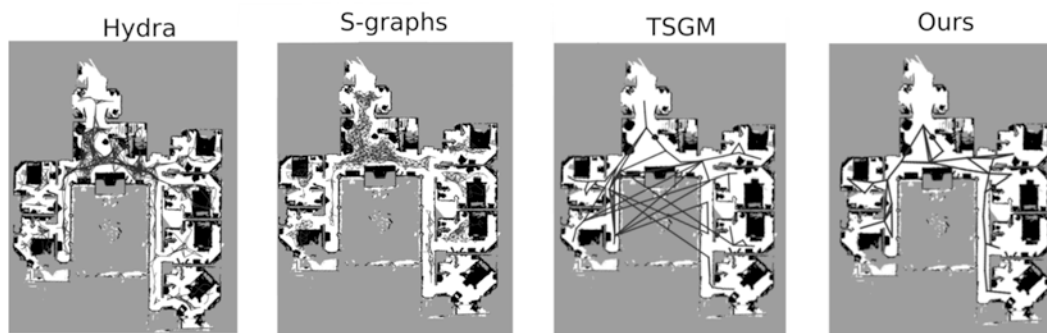


Рис. 6. Примеры графов локаций, построенных предложенным методом (Ours) и другими современными методами: Hydra, S-graphs, TSGM

лазерным сканером (лидаром), а также RGB-D камерами с видом вперед и назад. В ходе эксперимента робот двигался по предварительно заданному маршруту, проходящему по коридору длиной 100 м в обе стороны. Управление движением осуществлялось вручную. На вход методу подавались облака точек с лидара, а также данные лидарной одометрии, полученные алгоритмом Cartographer [4]. Данные с проезда робота (облака точек и одометрия по RGB-D камерам) подавались на вход методу в режиме реального времени.

Построенный предложенным методом граф в сравнении с двумерной картой занятости коридора и пройденной траекторией показан на Рис. 7. Несмотря на отсутствие динамических

объектов (люди, идущие рядом с роботом), разреженные облака точек с лидара и погрешности одометрии, метод успешно построил связный граф локаций, покрывающий весь коридор, отождествив коридор при прямом и обратном проезде. Видео с демонстрацией построения графа локаций на данных с робота доступно по ссылке <https://disk.yandex.ru/i/7tzfEeJ1lfBqcA>.

## Заключение

Предложенный метод поддерживает и обновляет граф локаций, а также локализуется в нем с использованием нейросетевого распознавания мест и классического двумерного сопоставления сканов. Эксперименты в симуляционных

Облако точек



Одометрия

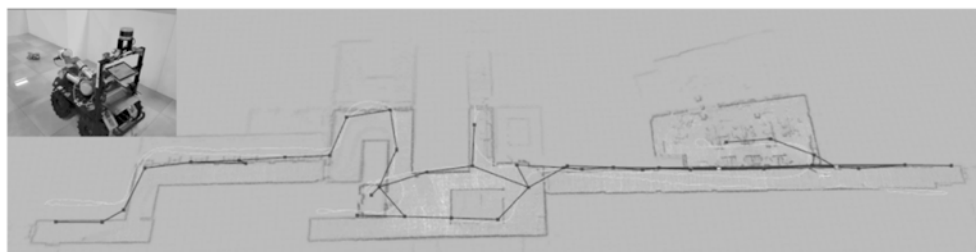


Рис. 7. Построенный граф локаций (линии – ребра, точки на линиях – вершины), в сравнении с двумерной картой среды (показана бледными контурами)



средах показали, что предложенный метод строит связные графы с низкой долей неконсистентных ребер и высокой площадью покрытия сцены, обеспечивающие высокую эффективность навигации. По навигационной эффективности метод значительно превосходит другие современные методы топологического картирования. Эксперименты, проведенные на данных с реального робота, показали работоспособность метода в реальном мире с разреженными облаками точек, неточной одометрией и динамическими объектами. В дальнейшем планируется добавление семантической информации в граф локаций для более полного представления сцены и более точной локализации, а также разработка методов навигации по графу локаций.

## Литература

1. Швец Е. А. Разработка моделей картирования и патрулирования коллективом беспилотных наземных роботов, использующих техническое зрение и эхолокацию: Автореф. Дисс. канд. техн. наук: [Место защиты: Институт проблем передачи информации им. А.А. Харкевича РАН]. – 2016. – 139 с //Режим доступа: <http://iitp.ru/upload/content/1327/synopsis.pdf>. – 2016.
2. Боковой А. В. Исследование методов одновременного картирования и локализации беспилотных летательных аппаратов по видеопотоку, полученному с единственной камеры //Второй Всероссийский научно-практический семинар "Беспилотные транспортные средства с элементами искусственного интеллекта". – 2015. – С. 26-33.
3. Labbé M., Michaud F. RTAB<sup>+</sup> Map as an open<sup>+</sup> source lidar and visual simultaneous localization and mapping library for large<sup>+</sup> scale and long<sup>+</sup> term online operation //Journal of field robotics. – 2019. – Vol. 36. – No. 2. – P. 416-446.
4. Hess W. et al. Real-time loop closure in 2D LIDAR SLAM //2016 IEEE international conference on robotics and automation (ICRA). – IEEE, 2016. – P. 1271-1278.
5. Muravyev K., Yakovlev K. Evaluation of RGB-D SLAM in large indoor environments //International Conference on Interactive Collaborative Robotics. – Cham: Springer International Publishing, 2022. – P. 93-104.
6. Щеголева Л. В., Воронов Р. В. Построение дорожного графа для маршрутизации мобильного робота в замкнутой системе коридоров //Инженерный вестник Дона. – 2015. – Т. 37. – №. 3. – С. 43.
7. Blochliger F. et al. Topomap: Topological mapping and navigation based on visual slam maps //2018 IEEE International Conference on Robotics and Automation (ICRA). – IEEE, 2018. – P. 3818-3825.
8. Gomez C. et al. Hybrid topological and 3d dense mapping through autonomous exploration for large indoor environments //2020 IEEE International Conference on Robotics and Automation (ICRA). – IEEE, 2020. – P. 9673-9679.
9. Schmid L. et al. A unified approach for autonomous volumetric exploration of large scale environments under severe odometry drift //IEEE Robotics and Automation Letters. – 2021. – Vol. 6. – No. 3. – P. 4504-4511.
10. Chen X. et al. Fast 3D sparse topological skeleton graph generation for mobile robot global planning //2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). – IEEE, 2022. – P. 10283-10289.
11. Hughes N., Chang Y., Carlone L. Hydra: A real-time spatial perception system for 3D scene graph construction and optimization //arXiv preprint arXiv:2201.13360. – 2022.
12. Yuan Y., Schwertfeger S. Incrementally building topology graphs via distance maps //2019 IEEE International Conference on Real-time Computing and Robotics (RCAR). – IEEE, 2019. – P. 468-474.
13. Kim N. et al. Topological semantic graph memory for image-goal navigation //Conference on Robot Learning. – PMLR, 2023. – P. 393-402.
14. Kwon O. et al. Visual graph memory with unsupervised representation for visual navigation //Proceedings of the IEEE/CVF International Conference on Computer Vision. – 2021. – P. 15890-15899.
15. Wiyatno R. R., Xu A., Paull L. Lifelong topological visual navigation //IEEE Robotics and Automation Letters. – 2022. – Vol. 7. – No. 4. – P. 9271-9278.
16. Yin P. et al. General place recognition survey: Towards the real-world autonomy age //arXiv preprint arXiv:2209.04497. – 2022.
17. Muravyev K., Yakovlev K. Evaluation of Topological Mapping Methods in Indoor Environments //IEEE Access. – 2023. – Vol. 11. – P. 132683-132698.
18. Muravyev K. et al. PRISM-TopoMap: Online Topological Mapping with Place Recognition and Scan Matching //arXiv preprint arXiv:2404.01674. – 2024.
19. Harris C. et al. A combined corner and edge detector //Alvey vision conference. – 1988. – Vol. 15. – No. 50. – P. 10-5244.
20. Rublee E. et al. ORB: An efficient alternative to SIFT or SURF //2011 International conference on computer vision. – IEEE, 2011. – P. 2564-2571.
21. Muja M., Lowe D. Flann-fast library for approximate nearest neighbors user manual //Computer Science Department, University of British Columbia, Vancouver, BC, Canada. – 2009. – Vol. 5. – P. 6.
22. Komorowski J. Minkloc3d: Point cloud based large-scale place recognition //Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. – 2021. – C. 1790-1799.
23. Chang A. et al. Matterport3d: Learning from rgb-d data in indoor environments //arXiv preprint arXiv:1709.06158. – 2017.
24. Bavlle H. et al. S-graphs<sup>+</sup>: Real-time localization and mapping leveraging hierarchical representations //arXiv preprint arXiv:2212.11770. – 2022.

**Муравьев Кирилл Федорович.** Федеральный исследовательский центр "Информатика и управление" РАН, Москва, Россия. Младший научный сотрудник. Область научных интересов: одновременное картирование и локализация (SLAM), навигация мобильных роботов, компьютерное зрение. E-mail: [muraviev@isa.ru](mailto:muraviev@isa.ru).

## Indoor Topological Mapping with Place Recognition and Scan Matching

K. F. Muravyev

Federal Research Center "Computer Science and Control" of Russian Academy of Sciences, Moscow, Russia

**Abstract.** Map building is one of the key tasks of autonomous mobile robots' navigation. Traditional mapping methods build dense metric map (e.g. as an occupancy grid). Maintaining such map in case of long-term navigation is difficult because of high computational costs and odometry error accumulation. Representing the environment as a sparse topological structure (e.g. a graph of locations) lets us eliminate these drawbacks and provide fast path planning. In this work, we propose a topological mapping method which builds and updates a graph of locations without use of global metric coordinates. For localization, the proposed method uses neural network-based place recognition in pair with 2D projection-based scan matching. We carry out experiments with our method in several photorealistic simulated scenes and on data from a real robot. In simulation, we compare our method with some state-of-the-art topological mapping methods. According to the results, the proposed method significantly outperforms competitors in terms of navigational efficiency, keeping graph connectivity, high scene coverage, and low part of inconsistent edges.

**Keywords:** simultaneous localization and mapping (SLAM), topological map, mobile robots.

DOI 10.14357/20718632240303      EDN CFSQSG

## References

1. Шве́ц Е. А. Разработка моделей картирования и патрулирования коллективом беспилотных наземных роботов, использующих техническое зрение и эхолокацию: Автореф. дисс. канд. техн. наук: [Место защиты: Институт проблем передачи информации им. АА Харкевича РАН]. – 2016. – 139 с //Режим доступа: <http://itp.ru/upload/content/1327/synopsis.pdf>. – 2016.
2. Боковой А. В. Исследование методов одновременного картирования и локализации беспилотных летательных аппаратов по видеопотоку, полученному с единственной камеры //Второй Всероссийский научно-практический семинар "Беспилотные транспортные средства с элементами искусственного интеллекта". – 2015. – С. 26-33.
3. Labbé M., Michaud F. RTAB-M ap as an open~s ource lidar and visual simultaneous localization and mapping library for large~scale and long~term online operation //Journal of field robotics. – 2019. – Vol. 36. – No. 2. – P. 416-446.
4. Hess W. et al. Real-time loop closure in 2D LIDAR SLAM //2016 IEEE international conference on robotics and automation (ICRA). – IEEE, 2016. – P. 1271-1278.
5. Muravyev K., Yakovlev K. Evaluation of RGB-D SLAM in large indoor environments //International Conference on Interactive Collaborative Robotics. – Cham: Springer International Publishing, 2022. – P. 93-104.
6. Щеголева Л. В., Воронов Р. В. Построение дорожного графа для маршрутизации мобильного робота в замкнутой системе коридоров //Инженерный вестник Дона. – 2015. – Т. 37. – №. 3. – С. 43.
7. Blochliger F. et al. Topomap: Topological mapping and navigation based on visual slam maps //2018 IEEE International Conference on Robotics and Automation (ICRA). – IEEE, 2018. – P. 3818-3825.
8. Gomez C. et al. Hybrid topological and 3d dense mapping through autonomous exploration for large indoor environments //2020 IEEE International Conference on Robotics and Automation (ICRA). – IEEE, 2020. – P. 9673-9679.
9. Schmid L. et al. A unified approach for autonomous volumetric exploration of large scale environments under severe odometry drift //IEEE Robotics and Automation Letters. – 2021. – Vol. 6. – No. 3. – P. 4504-4511.
10. Chen X. et al. Fast 3D sparse topological skeleton graph generation for mobile robot global planning //2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). – IEEE, 2022. – P. 10283-10289.
11. Hughes N., Chang Y., Carlone L. Hydra: A real-time spatial perception system for 3D scene graph construction and optimization //arXiv preprint arXiv:2201.13360. – 2022.
12. Yuan Y., Schwertfeger S. Incrementally building topology graphs via distance maps //2019 IEEE International Conference on Real-time Computing and Robotics (RCAR). – IEEE, 2019. – P. 468-474.
13. Kim N. et al. Topological semantic graph memory for image-goal navigation //Conference on Robot Learning. – PMLR, 2023. – P. 393-402.
14. Kwon O. et al. Visual graph memory with unsupervised representation for visual navigation //Proceedings of the

- IEEE/CVF International Conference on Computer Vision. – 2021. – P. 15890-15899.
15. Wiyatno R. R., Xu A., Paull L. Lifelong topological visual navigation //IEEE Robotics and Automation Letters. – 2022. – Vol. 7. – No. 4. – P. 9271-9278.
16. Yin P. et al. General place recognition survey: Towards the real-world autonomy age //arXiv preprint arXiv:2209.04497. – 2022.
17. Muravyev K., Yakovlev K. Evaluation of Topological Mapping Methods in Indoor Environments //IEEE Access. – 2023. – Vol. 11. – P. 132683-132698.
18. Muravyev K. et al. PRISM-TopoMap: Online Topological Mapping with Place Recognition and Scan Matching //arXiv preprint arXiv:2404.01674. – 2024.
19. Harris C. et al. A combined corner and edge detector //Alvey vision conference. – 1988. – Vol. 15. – No. 50. – P. 10-5244.
20. Rublee E. et al. ORB: An efficient alternative to SIFT or SURF //2011 International conference on computer vision. – IEEE, 2011. – P. 2564-2571.
21. Muja M., Lowe D. Flann-fast library for approximate nearest neighbors user manual //Computer Science Department, University of British Columbia, Vancouver, BC, Canada. – 2009. – Vol. 5. – P. 6.
22. Komorowski J. Minkloc3d: Point cloud based large-scale place recognition //Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. – 2021. – C. 1790-1799.
23. Chang A. et al. Matterport3d: Learning from rgb-d data in indoor environments //arXiv preprint arXiv:1709.06158. – 2017.
24. Bavle H. et al. S-graphs+: Real-time localization and mapping leveraging hierarchical representations //arXiv preprint arXiv:2212.11770. – 2022.

**Muravyev Kirill F.** Federal Research Center “Computer Sciences and Control” of Russian Academy of Sciences, Moscow, 119333, Russia. E-mail: muraviev@isa.ru