Перевод: английский - русский - www.onlinedoctranslator.com

ORB-SLAM: универсальный и точный монокуляр Система SLAM

Рауль Мур-Арталь, JMM Montiel, *Член IEEE*и Хуан Д. Тардос, *Член IEEE*

***Абстрактный*- В этой статье представлена ORB-SLAM, функциональная монокулярная система одновременной локализации и картирования (SLAM), которая работает в реальном времени в небольших и больших помещениях и на открытом воздухе. Система устойчива к серьезным помехам от движения, позволяет закрывать и перемещать широкий базовый цикл, а также включает в себя полную автоматическую инициализацию. Основываясь на превосходных алгоритмах последних лет, мы разработали с нуля новую систему, которая использует одни и те же функции для всех задач SLAM: отслеживание, отображение, перемещение и замыкание цикла. Стратегия выживания наиболее подходящего, которая выбирает точки и ключевые кадры реконструкции, приводит к превосходной надежности и создает компактную и отслеживаемую карту, которая увеличивается только при изменении содержимого сцены, что позволяет работать в течение всей жизни. Мы представляем исчерпывающую оценку в 27 последовательностях из самых популярных наборов данных. ORB-SLAM обеспечивает беспрецедентную производительность по сравнению с другими современными монокулярными подходами SLAM. В интересах сообщества мы публикуем исходный код.**

***Индекс терминов*—Картирование на всю жизнь, локализация, монокулярное зрение, распознавание, одновременная локализация и отображение (SLAM).**

Я ЯВВЕДЕНИЕ

В Известно, что настройка UNDLE (BA) обеспечивает точные оценки местоположения камеры, а также разреженную геометрическую реконструкцию [1], [2], учитывая, что обеспечивается сильная сеть совпадений и хорошие начальные предположения. Долгое время такой подход считался недоступным для приложений реального времени, таких как одновременная визуальная локализация и отображение (визуальный SLAM). Визуальный SLAM имеет цель оценить траекторию камеры при реконструкции окружающей среды. Теперь мы знаем, что для достижения точных результатов при непомерно высоких вычислительных затратах алгоритм SLAM в реальном времени должен предоставлять BA следующее.

1) Соответствующие наблюдения за функциями сцены (точки карты) среди подмножества выбранных кадров (ключевых кадров).
2) Поскольку сложность растет с увеличением количества ключевых кадров, их выбор должен избегать ненужной избыточности.
3) Сильная сетевая конфигурация ключевых кадров и точек для получения точных результатов, то есть хорошо распределенный набор ключевых кадров, наблюдающих точки со значительным параллаксом и с большим количеством совпадений замыкания цикла.

Рукопись получена 28 апреля 2015 г .; принята к печати 27 июля 2015 г. Дата публикации 24 августа 2015 г .; дата текущей версии 30 сентября 2015 г. Эта статья была рекомендована к публикации младшим редактором Д. Скарамуцца и редактором Д. Фоксом после оценки комментариев рецензентов. Эта работа была поддержана Генеральным директором исследований Испании в рамках проекта DPI2012-32168, стипендией Министерства образования FPU13 / 04175 и стипендией Гобьерно де Арагона B121 / 13.

Авторы из Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, 50018 Сарагоса, Испания (электронная почта: raulmur@unizar.es ; josemari@unizar.es ; tardos@unizar.es).

Цветные версии одного или нескольких рисунков в этом документе доступны на сайте http://ieeexplore.ieee.org.

Идентификатор цифрового объекта 10.1109 / TRO.2015.2463671

4) Первоначальная оценка положений ключевых кадров и местоположений точек для нелинейной оптимизации.
5) Локальная карта в исследовании, где оптимизация направлена на достижение масштабируемости.
6) Возможность выполнять быструю глобальную оптимизацию (например, граф позы) для закрытия циклов в реальном времени.

Первым приложением БА в реальном времени была работа Мурагона по визуальной одометрии. *и другие.* [3], за которой последовала новаторская работа Кляйна и Мюррея по SLAM [4], известная как параллельное отслеживание и отображение (PTAM). Этот алгоритм, хотя и ограничен мелкомасштабной операцией, предоставляет простые, но эффективные методы для выбора ключевых кадров, сопоставления функций, точечной триангуляции, локализации камеры для каждого кадра и перемещения после сбоя отслеживания. К сожалению, несколько факторов серьезно ограничивают его применение: отсутствие замыкания цикла и адекватной обработки окклюзий, низкая инвариантность точки зрения перемещения и необходимость вмешательства человека для начальной загрузки карты.

В этом исследовании мы опираемся на основные идеи PTAM, работу по распознаванию мест Гальвеса-Лопеса и Тардоса [5], масштабное замыкание цикла Strasdat. *и другие.* [6], а также использование информации о совместимости для крупномасштабных операций [7], [8], чтобы разработать с нуля ORB-SLAM, т. Е. Новую монокулярную систему SLAM, основные вклады которой заключаются в следующем.

1) Использование одних и тех же функций для всех задач: отслеживание, отображение, перемещение и закрытие цикла. Это делает нашу систему более эффективной, простой и надежной. Мы используем функции ORB [9], которые позволяют работать в реальном времени без графических процессоров, обеспечивая хорошую инвариантность к изменениям точки обзора и освещения.
2) Работа в режиме реального времени в больших средах. Благодаря использованию графа совместимости, отслеживание и отображение сосредоточены в локальной видимой области, независимо от размера глобальной карты.
3) Закрытие цикла в реальном времени на основе оптимизации графа позы, который мы называем *Essential Graph*. Он строится из остовного дерева, поддерживаемого системой, ссылок замыкания цикла и сильных ребер из графа ковидимости.
4) Перемещение камеры в реальном времени со значительной инвариантностью к точке обзора и освещению. Это позволяет восстановиться после сбоя отслеживания, а также улучшает повторное использование карты.
5) Новая автоматическая и надежная процедура инициализации, основанная на выборе модели, которая позволяет создавать начальную карту плоских и неплоских сцен.
6) А *выживание сильнейшего* подход к выбору точки карты и ключевого кадра, который щедр при порождении, но очень ограничен при отсечении. Эта политика улучшает надежность отслеживания и увеличивает срок службы, поскольку отбрасываются избыточные ключевые кадры.

1552-3098 © 2015 IEEE. Разрешено личное использование, но для переиздания / распространения требуется разрешение IEEE.
См. Http://www.ieee.org/publications Standards / публикации / права / index.html для получения дополнительной информации.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

Мы представляем обширную оценку в популярных общедоступных наборах данных для внутренней и внешней среды, включая ручные, автомобильные и роботизированные последовательности. Примечательно, что мы достигаем более высокой точности локализации камеры, чем современные методы в прямых методах [10], которые оптимизируют непосредственно по интенсивности пикселей вместо ошибок перепроецирования признаков. Мы включаем обсуждение в Раздел IX-B возможных причин, которые могут сделать методы, основанные на функциях, более точными, чем прямые методы.

Представленные здесь методы закрытия и перемещения цикла основаны на нашей предыдущей работе [11]. Предварительная версия системы представлена в [12]. В данной статье мы добавляем метод инициализации, *Essential Graph*, и совершенствовать все задействованные методы. Мы также подробно описываем все строительные блоки и проводим исчерпывающую экспериментальную проверку.

Насколько нам известно, это наиболее полное и надежное решение для монокулярного SLAM, и в интересах сообщества мы публикуем исходный код. Демонстрационные видеоролики и код можно найти на веб-странице нашего проекта.[1]

II. рВ восторге WORK

A. Признание места

Опрос Вильямса *и другие.* [13] сравнили несколько подходов к распознаванию мест и пришли к выводу, что методы, основанные на внешнем виде, то есть сопоставление изображения с изображением, лучше масштабируются в больших средах, чем методы «карта-карта» или «изображение-карта». В методах, основанных на внешнем виде, на передний план выходят методы мешков слов [14], такие как вероятностный подход FAB-MAP [15], из-за их высокой эффективности. DBoW2 [5] впервые использовал пакеты двоичных слов, полученные из дескрипторов BRIEF [16], вместе с очень эффективным детектором признаков FAST [17]. Это сократило более чем на порядок время, необходимое для извлечения признаков, по сравнению с функциями SURF [18] и SIFT [19], которые до сих пор использовались в подходах к пакетам слов. Хотя система продемонстрировала свою очень эффективную и надежную работу, использование BRIEF, не инвариантного по отношению к вращению или масштабированию, ограничили систему траекториями в плоскости и обнаружением петель с аналогичных точек зрения. В нашей предыдущей работе [11] мы предложили пакет распознавания места слов, построенный на DBoW2 с ORB [9]. ORB - это двоичные функции, инвариантные к вращению и масштабированию (в определенном диапазоне), что приводит к очень быстрому распознаванию с хорошей инвариантностью к точке обзора. Мы продемонстрировали высокую отзывчивость и надежность распознавателя в четырех различных наборах данных, которым требуется менее 39 мс (включая извлечение признаков) для извлечения кандидата цикла из базы данных изображений размером 10 КБ. В этом исследовании мы используем улучшенную версию этого распознавателя места, используя информацию о видимости и возвращая несколько гипотез при запросе к базе данных, а не только наилучшее совпадение. мы предложили пакет распознавания места слов, построенный на DBoW2 с ORB [9]. ORB - это двоичные функции, инвариантные к вращению и масштабированию (в определенном диапазоне), что приводит к очень быстрому распознаванию с хорошей инвариантностью к точке обзора. Мы продемонстрировали высокую отзывчивость и надежность распознавателя в четырех различных наборах данных, которым требуется менее 39 мс (включая извлечение признаков) для извлечения кандидата цикла из базы данных изображений размером 10 КБ. В этом исследовании мы используем улучшенную версию этого распознавателя места, используя информацию о видимости и возвращая несколько гипотез при запросе к базе данных, а не только наилучшее совпадение. мы предложили пакет распознавания места слов, построенный на DBoW2 с ORB [9]. ORB - это двоичные функции, инвариантные к вращению и масштабированию (в определенном диапазоне), что приводит к очень быстрому распознаванию с хорошей инвариантностью к точке обзора. Мы продемонстрировали высокую отзывчивость и надежность распознавателя в четырех различных наборах данных, которым требуется менее 39 мс (включая извлечение признаков) для извлечения кандидата цикла из базы данных изображений размером 10 КБ. В этом исследовании мы используем улучшенную версию этого распознава

Б. Инициализация карты

Монокуляр SLAM требует процедуры для создания начальной карты, потому что глубина не может быть восстановлена из одного изображения. Один из способов решения проблемы - изначально отслеживать известную структуру [20]. В контексте подходов к фильтрации точки могут быть инициализированы с высокой неопределенностью по глубине с использованием параметризации обратной глубины [21], которые, надеюсь, позже сойдутся к их реальным положениям. Недавнее полууплотненное произведение Энгеля. *и другие.* [10] следует аналогичному подходу, инициализируя глубину пикселей случайным значением с высокой дисперсией.

Методы инициализации из двух представлений либо предполагают локальную планарность сцены [4], [22] и восстанавливают относительную позу камеры из гомографии, используя метод Фогераса и Люстмана [23], либо вычисляют существенную матрицу [24], [25], которая моделирует плоские и общие сцены, используя пятибалльный алгоритм Нистера [26], который требует иметь дело с несколькими решениями. Оба метода реконструкции плохо ограничиваются при низком параллаксе и страдают от двоякой неоднозначности решения, если все точки плоской сцены находятся ближе к одному из центров камеры [27]. С другой стороны, если неплоская сцена видна с параллаксом, уникальная фундаментальная матрица может быть вычислена с помощью алгоритма из восьми точек [2], и относительная поза камеры может быть восстановлена без двусмысленности.

В Разделе IV мы представляем новый автоматический подход, основанный на выборе модели между гомографией для плоских сцен и фундаментальной матрицей для неплоских сцен. Статистический подход к выбору модели был предложен Торром.*и другие.* [28]. Исходя из аналогичных соображений, мы разработали алгоритм эвристической инициализации, который учитывает риск выбора фундаментальной матрицы в случаях, близких к вырожденным (т. Е. Планарным, почти плоским и низким параллаксом), в пользу выбора гомографии. В плоском случае в целях безопасности мы воздерживаемся от инициализации, если решение имеет двоякую неоднозначность, поскольку может быть выбрано поврежденное решение. Мы откладываем инициализацию до тех пор, пока метод не даст уникальное решение со значительным параллаксом.

C. Одновременная локализация и картографирование монокуляра

Монокулярный SLAM был первоначально решен путем фильтрации [20], [21], [29], [30]. При таком подходе каждый кадр обрабатывается фильтром для совместной оценки местоположения объектов карты и положения камеры. Он имеет недостатки, заключающиеся в бесполезной трате вычислений при обработке последовательных кадров с небольшим количеством новой информации и накоплении ошибок линеаризации. С другой стороны, подходы на основе ключевых кадров [3], [4] оценивают карту, используя только выбранные кадры (ключевые кадры), что позволяет выполнять более дорогостоящую, но точную оптимизацию BA, поскольку отображение не привязано к частоте кадров. Страсдат *и другие.* [31] продемонстрировали, что методы, основанные на ключевых кадрах, более точны, чем фильтрация, при тех же вычислительных затратах.

Наиболее представительной системой SLAM на основе ключевых кадров, вероятно, является PTAM Кляйна и Мюррея [4]. Это была первая работа, в которой была представлена идея разделения отслеживания и отображения камеры в параллельных потоках, и она продемонстрировала свою успешность для приложений дополненной реальности в реальном времени в небольших средах. Первоначальная версия была позже улучшена за счет функций ребер, шага оценки вращения во время отслеживания и лучшего метода перемещения [32]. Точки карты PTAM соответствуют углам FAST, сопоставленным с помощью корреляции фрагментов. Это делает точки полезными только для отслеживания, но не для распознавания места. Фактически, PTAM не обнаруживает большие петли, и перемещение основано на корреляции эскизов ключевых кадров с низким разрешением, что обеспечивает низкую инвариантность к точке обзора.

---

[1] http://webdiis.unizar.es/~раулмур / орбслам

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

Страсдат *и другие.* [6] представила крупномасштабную монокулярную систему SLAM с интерфейсом, основанным на оптическом потоке, реализованном на графическом процессоре, с последующим сопоставлением функций FAST и *только движение BA*, и бэкэнд, основанный на БА со скользящим окном. Замыкание петель было решено с помощью оптимизации графа позы с ограничениями сходства [7 степеней свободы (DoF)], что позволило скорректировать дрейф масштаба, появляющийся в монокулярном SLAM. Из этой работы мы берем идею закрытия цикла с помощью оптимизации графа позы с 7 степенями свободы и применяем ее к*Essential Graph* определено в Разделе III-D.

Страсдат *и другие.* [7] использовал интерфейс PTAM, но выполнял отслеживание только на локальной карте, полученной из графа ковидимости. Они предложили серверную часть оптимизации с двойным окном, которая непрерывно выполняет BA во внутреннем окне и создает график во внешнем окне ограниченного размера. Однако закрытие цикла эффективно только в том случае, если размер внешнего окна достаточно велик, чтобы охватить весь цикл. В нашей системе мы используем отличные идеи использования локальной карты, основанной на совмещаемости, и построения графа поз на основе графа совидимости, но применяем их в полностью переработанном интерфейсе и на стороне интерфейса. Другое отличие состоит в том, что вместо использования определенных функций для обнаружения петель (SURF) мы выполняем распознавание места для тех же отслеживаемых и сопоставленных функций, получая надежную перераспределение частоты кадров и обнаружение петель.

Пиркер *и другие.* [33] предложил CD-SLAM, т. Е. Очень полную систему, включающую замыкание цикла, перемещение по локализации, крупномасштабные операции и усилия по работе в динамических средах. Однако инициализация карты не упоминается. Отсутствие общедоступной реализации не позволяет нам проводить сравнение точности, надежности или крупномасштабных возможностей.

Визуальная одометрия песни *и другие.* [34] использует функции ORB для отслеживания и серверную часть BA со скользящим временным окном. Для сравнения, наша система является более общей, поскольку в них нет глобального перемещения, закрытия цикла и повторного использования карты. Они также используют известное расстояние от камеры до земли, чтобы ограничить смещение шкалы монокуляра.

Lim *и другие.* [25], работа, опубликованная после того, как мы представили нашу предварительную версию этой работы [12], также используют те же функции для отслеживания, отображения и обнаружения петель. Однако выбор BRIEF ограничивает систему траекториями в плоскости. Их система отслеживает только точки из последнего ключевого кадра; поэтому карта не используется повторно при повторном посещении (аналогично визуальной одометрии) и имеет проблему неограниченного роста. Мы качественно сравниваем наши результаты с этим подходом в Разделе VIII-E.

Недавняя работа Энгеля *и другие.* [10], известная как LSD-SLAM, может строить крупномасштабные полуидентичные карты, используя прямые методы (т. Е. Оптимизацию непосредственно по интенсивности пикселей изображения) вместо BA по объектам. Их результаты очень впечатляют, поскольку система способна работать в режиме реального времени без ускорения графического процессора, создавая полупрозрачную карту с большим количеством потенциальных приложений для робототехники, чем разреженный вывод, генерируемый функционально-ориентированным SLAM. Тем не менее, им по-прежнему нужны функции для обнаружения петель, и их точность определения местоположения камеры значительно ниже, чем в нашей системе и PTAM, как мы экспериментально показали в Разделе VIII-B. Этот удивительный результат обсуждается в разделе IX-B.

На полпути между прямым и функциональным методами находится полупрямая визуальная одометрия SVO Форстера. *и другие.* [22]. С участием-не требуя извлечения функций в каждом кадре, они могут работать с высокой частотой кадров, достигая впечатляющих результатов в квадракоптерах. Однако обнаружение петель не выполняется, и текущая реализация в основном рассчитана на камеры, смотрящие вниз.

Наконец, мы хотим обсудить выбор ключевых кадров. Все работы по визуальному SLAM в литературе согласны с тем, что выполнение BA со всеми точками и всеми кадрами невозможно. Работа Страсдата*и другие.* [31] показал, что наиболее экономичный подход - сохранить как можно больше точек, сохраняя только неизбыточные ключевые кадры. Подход PTAM заключался в очень осторожной вставке ключевых кадров, чтобы избежать чрезмерного роста вычислительной сложности. Эта ограничительная политика вставки ключевых кадров приводит к сбою отслеживания в сложных условиях исследования. Наш*выживание сильнейшего* Стратегия обеспечивает беспрецедентную надежность в сложных сценариях за счет максимально быстрой вставки ключевых кадров и последующего удаления избыточных кадров во избежание дополнительных затрат.

III. Систем Ообзор

A. Выбор функции

Одна из основных дизайнерских идей в нашей системе заключается в том, что те же функции, которые используются при отображении и отслеживании, используются для распознавания места, чтобы выполнить перемещение частоты кадров и обнаружение петель. Это делает нашу систему эффективной и избавляет от необходимости интерполировать глубину признаков распознавания по признакам, близким к SLAM, как в предыдущих работах [6], [7]. Нам требуются функции, которые требуют извлечения менее 33 мс на изображение, за исключением популярного SIFT (~300 мс) [19], SURF (~300 мс) [18], или недавний A-KAZE (~100 мс) [35]. Чтобы получить общие возможности распознавания мест, нам требуется инвариантность вращения, которая исключает BRIEF [16] и LDB [36].

Мы выбрали ORB [9], которые ориентированы на многомасштабные углы FAST с соответствующим 256-битным дескриптором. Их очень быстро вычислить и сопоставить, но при этом они хорошо инвариантны к точке обзора. Это позволяет нам сопоставлять их с широкими базами, повышая точность БА. Мы уже показали хорошую производительность ORB для распознавания мест в [11]. Хотя наша текущая реализация использует ORB, предлагаемые методы не ограничиваются этими функциями.

B. Три потока: отслеживание, локальное сопоставление и замыкание цикла

Наша система (см. Обзор на рис. 1) включает три потока, которые выполняются параллельно: отслеживание, локальное сопоставление и закрытие цикла. Отслеживание отвечает за локализацию камеры для каждого кадра и решение, когда вставлять новый ключевой кадр. Сначала мы выполняем начальное сопоставление функции с предыдущим кадром и оптимизируем позу, используя*только движение BA.* Если отслеживание потеряно (например, из-за окклюзии или резких движений), модуль распознавания места используется для выполнения глобального перемещения. После первоначальной оценки положения камеры и сопоставления характеристик извлекается локальная видимая карта с использованием графа видимости ключевых кадров, поддерживаемого системой [см. Рис. 2 (a) и (b)]. Тогда совпадения с местными точками карты

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

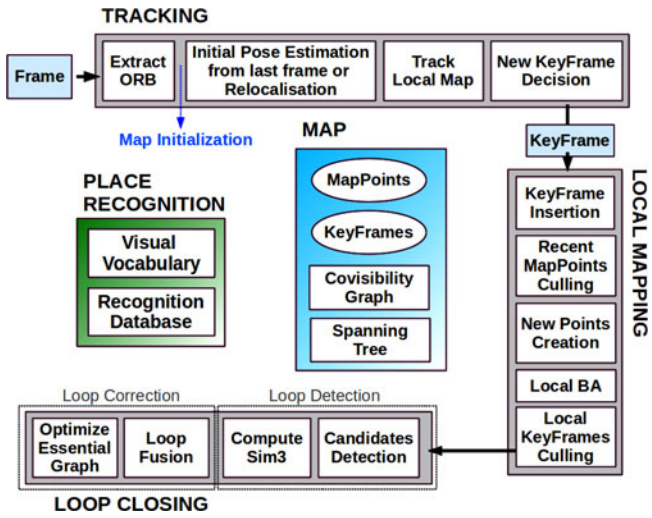

Рис. 1. Обзор системы ORB-SLAM, показывающий все шаги, выполняемые потоками отслеживания, локального сопоставления и закрытия цикла. Также показаны основные компоненты модуля распознавания мест и карта.

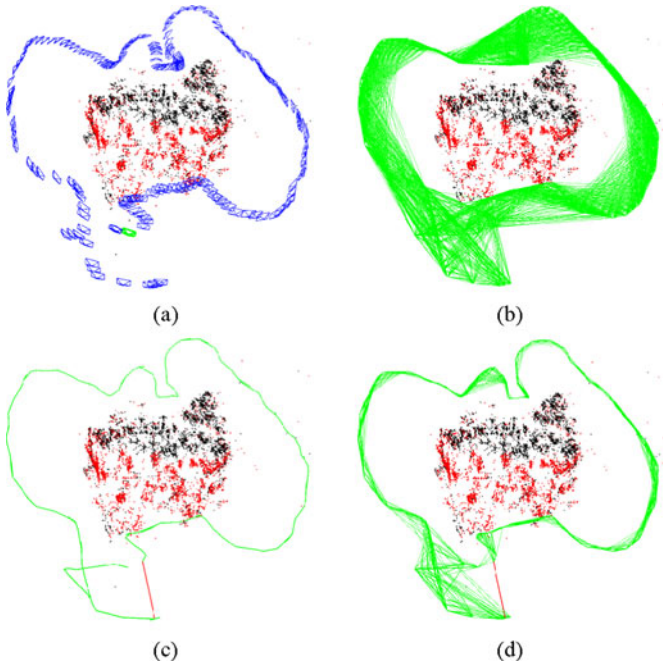

Рис. 2. Реконструкция и графики в последовательности *fr3 долго из домашнего хозяйства* из теста TUM RGB-D [38]. (a) Ключевые кадры (синий), текущая камера (зеленый), точки карты (черный, красный), текущие локальные точки карты (красный). (б) Граф ковидимости. (c) Остовное дерево (зеленый) и замыкание петли (красный). (d) Существенный граф.

поиск выполняется путем перепроецирования, а поза камеры снова оптимизируется со всеми совпадениями. Наконец, поток отслеживания решает, вставлен ли новый ключевой кадр. Все шаги отслеживания подробно описаны в Разделе V. Новая процедура создания начальной карты представлена в Разделе IV.

Локальное сопоставление обрабатывает новые ключевые кадры и выполняет *местный БА* для достижения оптимальной реконструкции антуража позы камеры. Новые соответствия для несогласованного ORB в новом ключевом кадре ищутся в связанных ключевых кадрах в графе ковидимости для триангуляции новых точек. Через некоторое время после создания на основе информации, собранной во время отслеживания, применяется строгая политика отбраковки точек, чтобы сохранить только точки высокого качества. Локальное сопоставление также отвечает за отбраковку избыточных ключевых кадров. Мы подробно объясняем все этапы локального сопоставления в Разделе VI.

Замыкание цикла ищет петли с каждым новым ключевым кадром. Если петля обнаружена, мы вычисляем преобразование подобия, которое сообщает о дрейфе, накопленном в петле. Затем выравниваются обе стороны петли и соединяются повторяющиеся точки. Наконец, для достижения глобальной согласованности выполняется оптимизация графа позы по ограничениям сходства [6]. Главное новшество в том, что мы проводим оптимизацию над *Essential Graph*, т. е. более разреженный подграф графа совместимости, который объясняется в Разделе III-D. Этапы обнаружения и исправления петель подробно описаны в Разделе VII.

Мы используем алгоритм Левенберга – Марквардта, реализованный в g2o [37], для проведения всех оптимизаций. В Приложении мы описываем условия ошибок, функции затрат и переменные, участвующие в каждой оптимизации.

C. Точки карты, ключевые кадры и их выбор

Каждая точка карты $п_я$ хранит следующее:

1) его трехмерное положение $\mathbf{Икс}_{ш,\,я}$ в мировой системе координат;
2) направление взгляда $\mathbf{п}_я$, который является средним единичным вектором всех его направлений обзора (лучи, соединяющие точка с оптическим центром наблюдающих ее ключевых кадров);
3) репрезентативный дескриптор ORB $\mathbf{D}_я$, который является ассоциированным дескриптором ORB, расстояние Хэмминга которого минимально по сравнению со всеми другими ассоциированными дескрипторами в ключевых кадрах, в которых наблюдается точка;
4) максимум $d_{Максимум}$ и минимум $d_{мин}$ расстояния, на которых можно наблюдать точку, в соответствии с пределами масштабной инвариантности функций ORB.

Каждый ключевой кадр $К_я$ хранит следующее:

1) поза камеры $\mathbf{T}_{iw}$, которое представляет собой преобразование твердого тела, которое переводит точки из мира в систему координат камеры;
2) характеристики камеры, включая фокусное расстояние и главную точку;
3) все объекты ORB, извлеченные в кадре, связанные или не связанные с точкой карты, координаты которой не искажаются, если предоставляется модель искажения.

Точки карты и ключевые кадры создаются с помощью обширной политики, в то время как более поздний очень требовательный механизм отбраковки отвечает за обнаружение избыточных ключевых кадров и неправильно сопоставленных или не отслеживаемых точек карты. Это позволяет гибко расширять карту во время исследования, что повышает надежность отслеживания в жестких условиях (например, вращение, быстрое перемещение), в то время как ее размер ограничен при постоянных пересмотрах в одну и ту же среду, т. Е. В течение всей жизни. Кроме того, наши карты содержат очень мало выбросов по сравнению с PTAM за счет меньшего количества точек. Процедуры отбраковки точек карты и ключевых кадров объясняются в разделах VI-B и VI-E, соответственно.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

D. Граф ковидимости и существенный граф

Информация о доступности между ключевыми кадрами очень полезна в нескольких задачах нашей системы и представлена в виде неориентированного взвешенного графа, как в [7]. Каждый узел является ключевым кадром, и существует граница между двумя ключевыми кадрами, если они совместно используют наблюдения одних и тех же точек карты (не менее 15), являясь весом θ края - количество общих точек карты.

Чтобы исправить цикл, мы выполняем оптимизацию графа позы [6], которая распределяет ошибку закрытия цикла по графу. Чтобы не включать все ребра, предоставляемые графом ковидимости, который может быть очень плотным, мы предлагаем построить *Essential Graph* который сохраняет все узлы (ключевые кадры), но меньше ребер, сохраняя при этом сильную сеть, дающую точные результаты. Система постепенно строит связующее дерево из начального ключевого кадра, которое обеспечивает связанный подграф графа совместимости с минимальным количеством ребер. Когда вставляется новый ключевой кадр, он включается в дерево, связанное с ключевым кадром, который разделяет большинство точечных наблюдений, а когда ключевой кадр стирается политикой отбраковки, система обновляет ссылки, затронутые этим ключевым кадром. В *Essential Graph* содержит остовное дерево, подмножество ребер из графа совместимости с высокой совместимостью ($\theta_{\text{мин}} = 100$), и края закрытия петли, в результате чего образуется прочная сеть камер. На рис. 2 показан пример графа совместимости, остовного дерева и связанного с ним существенного графа. Как показано в экспериментах Раздела VIII-E, при выполнении оптимизации графа позы решение настолько точное, что дополнительная полная оптимизация BA едва ли улучшает решение. Эффективность существенного графа и влияние $\theta_{\text{мин}}$ показан в конце Раздела VIII-E.

E. Мешки со словами "Распознавание мест"

В систему встроен модуль распознавания мест пакетов слов, основанный на DBoW2.[2] [5], чтобы выполнить обнаружение петель и перемещение. Визуальные слова - это просто дискретизация пространства дескрипторов, известная как визуальный словарь. Словарь создается в автономном режиме с дескрипторами ORB, извлеченными из большого набора изображений. Если изображения достаточно общие, один и тот же словарь может использоваться для разных сред, обеспечивая хорошую производительность, как показано в нашей предыдущей работе [11]. Система постепенно создает базу данных, которая содержит инвертированный индекс, в котором для каждого визуального слова в словаре хранятся ключевые кадры, в которых оно было замечено, так что запросы к базе данных могут выполняться очень эффективно. База данных также обновляется, когда ключевой кадр удаляется процедурой отбраковки.

Поскольку существует визуальное перекрытие между ключевыми кадрами, при запросе к базе данных не будет существовать уникальный ключевой кадр с высоким баллом. Исходный DBoW2 учел это перекрытие, суммируя количество изображений, близких по времени. Это ограничение не включает ключевые кадры, просматривающие одно и то же место, но вставленные в другое время. Вместо этого мы группируем те ключевые кадры, которые связаны в графе совместимости. Кроме того, наша база данных возвращает все совпадения ключевых кадров, чьи оценки выше, чем 75% лучший результат.

[2]https://github.com/dorian3d/DBoW2

О дополнительном преимуществе представления пакетов слов для сопоставления признаков было сообщено в [5]. Когда мы хотим вычислить соответствия между двумя наборами функций ORB, мы можем ограничить сопоставление методом грубой силы только теми функциями, которые принадлежат одному и тому же узлу в словарном дереве на определенном уровне (мы выбираем второй из шести), ускоряя вверх по поиску. Мы используем это *обманывать* при поиске совпадений для триангуляции новых точек, а также при обнаружении петель и перемещении. Мы также уточняем соответствия с помощью теста согласованности ориентации (подробности см. В [11]), который отбрасывает выбросы, обеспечивая когерентное вращение для всех соответствий.

IV. Автоматический Мар янитиализация

Цель инициализации карты - вычислить относительную позу между двумя кадрами для триангуляции начального набора точек карты. Этот метод не должен зависеть от сцены (плоской или общей) и не требует вмешательства человека для выбора хорошей конфигурации с двумя ракурсами, т. Е. Конфигурации со значительным параллаксом. Мы предлагаем вычислить параллельно две геометрические модели: гомографию, предполагающую плоскую сцену, и фундаментальную матрицу, предполагающую неплоскую сцену. Затем мы используем эвристику для выбора модели и пытаемся восстановить относительную позу с помощью определенного метода для выбранной модели. Наш метод инициализируется только тогда, когда он уверен, что конфигурация с двумя представлениями безопасна, обнаруживая случаи низкого параллакса и хорошо известную двукратную плоскую неоднозначность [27], избегая инициализации поврежденной карты. Шаги нашего алгоритма следующие.

1) *Найдите начальные соответствия:* Извлечь функции ORB (только в самом мелком масштабе) в текущем кадре F_c и ищи совпадения $\mathbf{Икс}_c \leftrightarrow \mathbf{Икс}_p$ в системе отсчета F_p. Если найдено недостаточно совпадений, сбросьте опорный кадр.
2) *Параллельное вычисление двух моделей:* Вычислить в параллельных потоках гомографию $\mathbf{ЧАС}_{cr}$ и фундаментальная матрица $\mathbf{F}_{cr}$ в виде

$$\mathbf{Икс}_c \text{ знак равно } \mathbf{ЧАС}_{cr}\mathbf{Икс}_p,\ \mathbf{Икс}_c^{\mathsf{T}}\mathbf{F}_{cr}\mathbf{Икс}_p \text{ знак равно } 0 \tag{1}$$

с нормализованным алгоритмом DLT и алгоритмом из восьми пунктов, соответственно, как объяснено в [2] внутри схемы RANSAC. Чтобы сделать процедуру однородной для обеих моделей, число итераций указывается в качестве префикса и одинаково для обеих моделей, а также точки, которые будут использоваться на каждой итерации: восемь для фундаментальной матрицы и четыре из них для гомографии. На каждой итерации мы вычисляем оценку S_M для каждой модели M ($ЧАС$ для омографии, F для фундаментальной матрицы)

$$S_M \text{ знак равно } \sum_i \left(\rho_M\left(d^2_{(\mathbf{Икс}_c^i, \mathbf{Икс}_p^i, M)}\right) + \rho_M\left(d^2_{rc}\left(\mathbf{Икс}_c^i, \mathbf{Икс}_p^i, M\right)\right)\right)$$

$$\rho_M(d^2 \text{ знак равно } \begin{cases} \Gamma - d^2, & \text{если } d^2 < T_M \\ 0, & \text{если } d^2 \geq T_M \end{cases} \tag{2}$$

где d^2_{cr} и d^2_{rc} являются симметричными ошибками передачи [2] от одного кадра к другому. T_M порог отклонения выбросов на основе χ^2 тест на 95% ($T_{ЧАС}$ знак равно 5.99, T_F знак равно 3.84, предполагая стандартное отклонение в 1 пиксель в

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

погрешность измерения). Г определяется как $T_{ЧАС}$ чтобы обе модели получили одинаковую оценку d в их начальной области, чтобы сделать процесс однородным.

Мы сохраняем гомографию и фундаментальную матрицу с наивысшим баллом. Если модель не может быть найдена (недостаточно вставок), мы снова перезапускаем процесс с шага 1.

3) *Выбор модели:* Если сцена плоская, почти плоская или имеется низкий параллакс, это можно объяснить гомографией. Однако фундаментальная матрица также может быть найдена, но проблема плохо ограничена [2], и любая попытка восстановить движение из фундаментальной матрицы приведет к неверным результатам. Мы должны выбрать гомографию, так как метод реконструкции будет правильно инициализироваться с плоскости, или он обнаружит случай низкого параллакса и откажется от инициализации. С другой стороны, неплоская сцена с достаточным параллаксом может быть объяснена только с помощью фундаментальной матрицы, но также можно найти гомографию, объясняющую подмножество совпадений, если они лежат на плоскости или имеют низкий параллакс (они далеко) . В этом случае следует выбрать фундаментальную матрицу. Мы обнаружили, что надежная эвристика предназначена для вычисления

$$p_{ЧАС} \text{ знак равно } \frac{S_{ЧАС}}{S_{ЧАС} + S_F} \tag{3}$$

и выберите омографию, если $p_{ЧАС} > 0.45$, который адекватно отражает случаи плоского и малого параллакса. В противном случае мы выбираем фундаментальную матрицу.

4) *Движение и структура после восстановления движения:* После выбора модели мы извлекаем связанные с ней гипотезы движения.

В случае гомографии мы восстанавливаем восемь гипотез движения, используя метод Фогераса и Люстмана [23]. Метод предлагает тесты на жизнеспособность для выбора действительного решения. Однако эти тесты терпят неудачу при низком параллаксе, поскольку точки легко проходят перед или за камерой, что может привести к выбору неправильного решения. Мы предлагаем провести прямую триангуляцию восьми решений и проверить, существует ли одно решение, в котором большинство точек видно с параллаксом, перед обеими камерами и с низкой ошибкой перепроецирования. Если нет явного решения-победителя, мы не инициализируем и продолжаем с шага 1. Этот метод устранения неоднозначности решений делает нашу инициализацию устойчивой при низком параллаксе и конфигурации двойной неоднозначности и может считаться ключом к надежности нашего метода.

В случае фундаментальной матрицы мы преобразуем ее в существенную матрицу, используя калибровочную матрицу **K** в виде

$$\mathbf{E}_{rc} \text{ знак равно } \mathbf{K}_T \mathbf{F}_{rc} \mathbf{K} \tag{4}$$

а затем получить четыре гипотезы движения с помощью метода разложения по сингулярным числам, описанного в [2]. Мы триангулируем четыре решения и выбираем реконструкцию, как это сделано для гомографии.

5) *Регулировка связки:* Наконец, мы выполняем *полный бакалавр* (подробности см. в Приложении), чтобы уточнить первоначальную реконструкцию.

Пример сложной инициализации в последовательности открытого робота NewCollege [39] показан на рис. 3. Можно увидеть, как PTAM и LSD-SLAM инициализировали все точки.

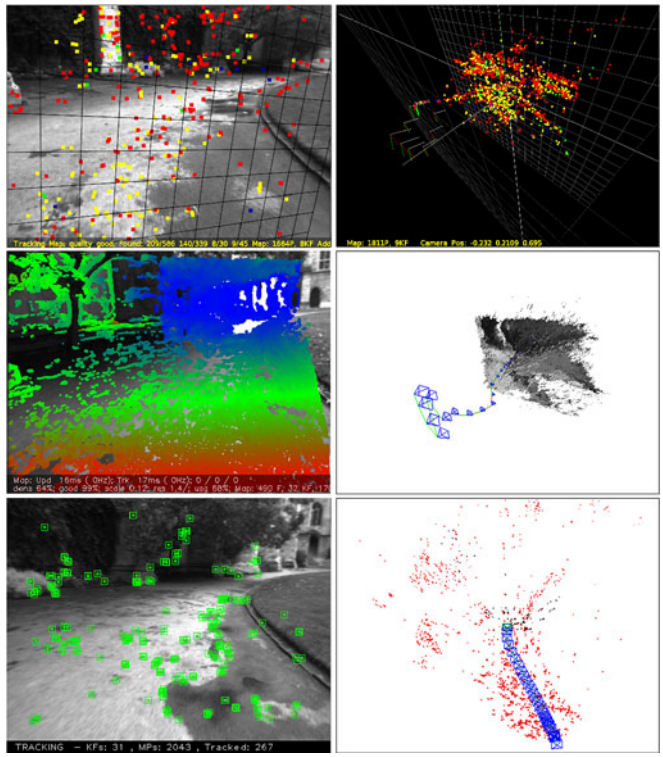

Рис 3. Вверху: PTAM, в центре: LSD-SLAM, внизу: ORB-SLAM, некоторое время после инициализации в последовательности NewCollege [39]. PTAM и LSD-SLAM инициализируют поврежденное плоское решение, в то время как наш метод автоматически инициализируется из фундаментальной матрицы, когда он обнаруживает достаточный параллакс. В зависимости от того, какие ключевые кадры выбраны вручную, PTAM также может хорошо инициализироваться.

в плоскости, в то время как наш метод ждал, пока не будет достаточного параллакса, правильно инициализируясь из фундаментальной матрицы.

B. Тстойка

В этом разделе мы описываем шаги потока отслеживания, которые выполняются с каждым кадром с камеры. Оптимизация позы камеры, упомянутая в нескольких шагах, состоит в *только движение BA*, который описан в Приложении.

A. Извлечение ORB

Мы извлекаем углы БЫСТРО на восьми уровнях шкалы с коэффициентом масштабирования 1,2. Для изображений с разрешением от512 × 384 к 752 × 480 пикселей, которые мы сочли подходящими для извлечения 1000 углов при более высоком разрешении, поскольку 1241 × 376 в наборе данных KITTI [40] мы извлекаем 2000 углов. Чтобы обеспечить однородное распределение, мы делим каждый масштабный уровень в сетке, пытаясь выделить не менее пяти углов на ячейку. Затем мы обнаруживаем углы в каждой ячейке, адаптируя порог детектора, если обнаружено недостаточно углов. Количество углов, сохраняемых на ячейку, также адаптируется, если некоторые ячейки не содержат углов (без текстуры или с низким контрастом). Затем для сохраненных углов FAST вычисляются ориентация и дескриптор ORB. Дескриптор ORB используется при сопоставлении всех функций, в отличие от поиска по корреляции патчей в PTAM.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

Б. Первоначальная оценка позы из предыдущего кадра

Если отслеживание было успешным для последнего кадра, мы используем модель движения с постоянной скоростью, чтобы предсказать положение камеры и выполнить управляемый поиск точек карты, наблюдаемых в последнем кадре. Если найдено недостаточно совпадений (т. Е. Модель движения явно нарушена), мы используем более широкий поиск точек карты вокруг их положения в последнем кадре. Затем поза оптимизируется с учетом найденных соответствий.

C. Первоначальная оценка позы посредством глобального перемещения

Если отслеживание потеряно, мы преобразуем кадр в пакет слов и запрашиваем в базе данных распознавания кандидатов ключевых кадров для глобального перемещения. Мы вычисляем соответствия с ORB, связанные с точками карты в каждом ключевом кадре, как объяснено в Разделе III-E. Затем мы альтернативно выполняем итерации RANSAC для каждого ключевого кадра и пытаемся найти позу камеры, используя алгоритм PnP [41]. Если мы находим позу камеры с достаточным количеством вставок, мы оптимизируем позу и выполняем управляемый поиск большего количества совпадений с точками карты кандидата ключевого кадра. Наконец, поза камеры снова оптимизируется, и, если поддерживается достаточным количеством вставок, процедура отслеживания продолжается.

D. Отслеживание местной карты

Как только у нас будет оценка положения камеры и начальный набор совпадений функций, мы можем спроецировать карту в кадр и искать больше соответствий точек карты. Чтобы ограничить сложность на больших картах, мы проецируем только локальную карту. Эта локальная карта содержит набор ключевых кадров K_1, которые разделяют точки карты с текущим кадром, и набор K_2 с соседями по ключевым кадрам K_1 в графе ковидимости. На локальной карте также есть опорный ключевой кадр. $K_{\text{ссылка}} \in K_1$, который разделяет большинство точек карты с текущим фреймом. Теперь каждая точка карты, видимая в K_1 и K_2 ищется в текущем кадре следующим образом.

1) Вычислить проекцию точки карты **Икс** в текущем кадре. Отменить, если он выходит за рамки изображения.
2) Вычислить угол между текущим лучом обзора. **v** и точка на карте означают направление взгляда **п**. Отменить, если $\mathbf{v} \cdot \mathbf{n} < \cos(60^\circ)$.
3) Вычислить расстояние d от точки карты до центра камеры. Отменить, если он находится за пределами области масштабной инвариантности точки карты $d \in [d_{\text{мин}}, d_{\text{максимум}}]$.
4) Вычислить масштаб кадра по соотношению $d / d_{\text{мин}}$.
5) Сравните репрезентативный дескриптор **D** точки карты с еще несопоставленными функциями ORB в кадре, в прогнозируемом масштабе и близком **Икс**и свяжите точку на карте с лучшим совпадением.

Поза камеры наконец оптимизирована со всеми точками карты, найденными в кадре.

E. Решение о новом ключевом кадре

Последний шаг - решить, будет ли текущий кадр порожден как новый ключевой кадр. Поскольку в локальном сопоставлении есть механизм для отсеивания избыточных ключевых кадров, мы постараемся вставлять ключевые кадры как можно быстрее, потому что это делает отслеживание более надежным для сложные движения камеры, обычно повороты. Чтобы вставить новый ключевой кадр, должны быть выполнены все следующие условия.

1) С момента последнего глобального перемещения должно пройти более 20 кадров.
2) Локальное сопоставление неактивно или с момента вставки последнего ключевого кадра прошло более 20 кадров.
3) Текущий кадр отслеживает не менее 50 точек.
4) Текущий кадр отслеживает менее 90% точек, чем $K_{\text{ссылка}}$. Вместо использования критерия расстояния до других ключевых кадров в качестве PTAM мы вводим минимальное визуальное изменение (условие 4). Условие 1 обеспечивает хорошее перемещение, а условие 3 - хорошее отслеживание. Если ключевой кадр вставлен, когда локальное отображение занято (вторая часть условия 2), отправляется сигнал, чтобы остановить локальный BA, чтобы он мог как можно скорее обработать новый ключевой кадр.

VI. Local Mприложение

В этом разделе мы описываем шаги, выполняемые локальным сопоставлением с каждым новым ключевым кадром. $K_я$.

A. Вставка ключевого кадра

Сначала мы обновляем граф ковидимости, добавляя новый узел для $K_я$и обновление краев, полученных из общих точек карты, с другими ключевыми кадрами. Затем мы обновляем связку связующего дерева. $K_я$с ключевым кадром с большинством общих точек. Затем мы вычисляем пакеты слов, представляющие ключевой кадр, что помогает в ассоциации данных для триангуляции новых точек.

Б. Удаление недавних точек карты

Для того, чтобы точки карты сохранялись на карте, они должны пройти ограничительный тест в течение первых трех ключевых кадров после создания, что гарантирует их отслеживание и отсутствие ошибочной триангуляции, т. Е. Из-за ложной ассоциации данных. Очко должно соответствовать этим двум условиям.

1) Отслеживание должно найти точку более чем в 25% кадров, в которых, как ожидается, она будет видна.
2) Если после создания точки карты прошло более одного ключевого кадра, он должен наблюдаться как минимум из трех ключевых кадров.

После того, как точка на карте прошла этот тест, ее можно удалить только в том случае, если в любой момент она наблюдается менее чем из трех ключевых кадров. Это может произойти, когда ключевые кадры отбираются, а локальный BA отбрасывает выбросы. Благодаря этой политике наша карта содержит очень мало выбросов.

C. Создание новой точки карты

Новые точки карты создаются путем триангуляции ORB из связанных ключевых кадров. K_cв графе ковидимости. За каждый непревзойденный ORB в$K_я$, мы ищем совпадение с другой несогласованной точкой в другом ключевом кадре. Это сопоставление выполняется, как описано в разделе III-E, и отбрасываются те совпадения, которые не соответствуют эпиполярному ограничению. Пары ORB триангулируются, и для принятия новых точек проверяется положительная глубина в обеих камерах, параллакс, ошибка перепроецирования и согласованность масштаба. Первоначально точка на карте наблюдается из двух ключевых кадров, но может быть

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

совпадает с другими; поэтому он проецируется в остальных связанных ключевых кадрах, и поиск соответствий осуществляется, как подробно описано в Разделе VD.

D. Регулировка локального пакета

В *местный БА* оптимизирует текущий обрабатываемый ключевой кадр $K_я$, все связанные с ним ключевые кадры в графе ковидимости K_C, и все точки карты, видимые этими ключевыми кадрами. Все другие ключевые кадры, которые видят эти точки, но не связаны с текущим обрабатываемым ключевым кадром, включаются в оптимизацию, но остаются фиксированными. Наблюдения, отмеченные как выбросы, отбрасываются в середине и в конце оптимизации. См. Приложение для более подробной информации об этой оптимизации.

E. Выбор локального ключевого кадра

Чтобы сохранить компактную реконструкцию, локальное сопоставление пытается обнаружить избыточные ключевые кадры и удалить их. Это выгодно, так как сложность BA растет с увеличением количества ключевых кадров, но также потому, что это позволяет работать в течение всей жизни в той же среде, поскольку количество ключевых кадров не будет неограниченно расти, если только визуальный контент в сцене не изменится. Мы отбрасываем все ключевые кадры вK_Cчьи 90% точек карты были замечены по крайней мере в трех других ключевых кадрах в том же или более мелком масштабе. Условие масштабирования гарантирует, что точки карты поддерживают ключевые кадры, по которым они измеряются с наибольшей точностью. Эта политика была вдохновлена политикой, предложенной в работе Тана. *и другие.* [24], где ключевые кадры были отброшены после процесса обнаружения изменений.

VII. Loop Спотеря

Нить закрытия цикла принимает $K_я$, последний ключевой кадр, обработанный локальным сопоставлением, и пытается обнаружить и закрыть петли. Далее описываются шаги.

A. Обнаружение кандидатов цикла

Сначала мы вычисляем сходство между вектором мешка слов $K_я$и все его соседи в графе совместимости ($\theta_{мин}$знак равно 30) и сохранить самый низкий балл $s_{мин}$. Затем мы запрашиваем базу данных распознавания и отбрасываем все те ключевые кадры, оценка которых ниже, чем $s_{мин}$. Это аналогичная операция для повышения устойчивости, что и нормализующий балл в DBoW2, который вычисляется из предыдущего изображения, но здесь мы используем информацию о совместимости. Кроме того, все эти ключевые кадры, напрямую связанные с$K_я$исключаются из результатов. Чтобы принять кандидата в цикл, мы должны последовательно обнаружить три согласованных кандидата в цикл (ключевые кадры, соединенные в графе совместимости). Может быть несколько кандидатов в петли, если есть несколько мест с похожим внешним видом на$K_я$.

Б. Вычислить преобразование подобия

В монокулярном SLAM есть семь степеней свободы, в которых карта может дрейфовать: три перевода, три поворота и масштабный коэффициент [6]. Следовательно, чтобы замкнуть цикл, нам нужно вычислить преобразование подобия из текущего ключевого кадра.$K_я$к ключевому кадру цикла $K_л$что сообщает нам о накопленной ошибке в петле. Вычисление этого подобия будет также служить геометрической проверкой петли.

Сначала мы вычисляем соответствия между ORB, связанным с точками карты в текущем ключевом кадре, и ключевыми кадрами-кандидатами в цикл, следуя процедуре, описанной в Разделе III-E. На данный момент у нас есть трехмерные соответствия для каждого кандидата в петли. В качестве альтернативы мы выполняем итерации RANSAC с каждым кандидатом, пытаясь найти преобразование подобия, используя метод Хорна [42]. Если мы найдем сходство$\mathbf{S}_{il}$при наличии достаточного количества вставок мы оптимизируем его (см. Приложение) и выполняем управляемый поиск большего количества соответствий. Оптимизируем еще раз, и если$\mathbf{S}_{il}$поддерживается достаточным количеством вставок, цикл с $K_л$принято.

C. Loop Fusion.

Первым шагом в исправлении цикла является объединение дублированных точек карты и вставка новых ребер в граф видимости, которые присоединят замыкание цикла. Во-первых, текущая поза ключевого кадра$\mathbf{T}_{iw}$исправлено преобразованием подобия $\mathbf{S}_{il}$, и эта поправка распространяется на всех соседей $K_я$, конкатенация преобразований, чтобы обе стороны цикла были выровнены. Все точки карты, видимые ключевым кадром цикла и его соседями, проецируются в$K_я$, и его соседи и совпадения ищутся в узкой области вокруг проекции, как это сделано в Разделе VD. Все эти точки на карте совпали, а те, которые были выбраны при вычислении$\mathbf{S}_{il}$слиты. Все ключевые кадры, участвующие в слиянии, обновят свои ребра в графе совместимости, эффективно создав ребра, которые присоединяют замыкание цикла.

D. Основная оптимизация графа

Чтобы эффективно замкнуть цикл, мы выполняем оптимизацию графа позы над *Essential Graph*, описанный в Разделе III-D, который распределяет ошибку закрытия цикла по графику. Оптимизация выполняется по преобразованиям подобия, чтобы исправить дрейф масштаба [6]. Условия ошибки и функция стоимости подробно описаны в Приложении. После оптимизации каждая точка карты преобразуется в соответствии с коррекцией одного из ключевых кадров, который ее наблюдает.

VIII. Experiments

Мы выполнили обширную экспериментальную проверку нашей системы в большой последовательности роботов NewCollege [39], оценивая общую производительность системы, в 16 портативных внутренних последовательностях теста TUM RGB-D [38], оценивая точность локализации, перебазирование и пожизненные возможности, а также в 10 сценах автомобилей вне помещения из набора данных KITTI [40], оценивающих крупномасштабные операции в реальном времени, точность локализации и эффективность оптимизации графа позы.

Наша система работает в режиме реального времени и обрабатывает изображения точно с той частотой кадров, с которой они были получены. Мы проводили все эксперименты с Intel Core i7-4700MQ (четыре ядра @ 2,40 ГГц) и 8 ГБ оперативной памяти. ORB-SLAM имеет три основных потока, которые выполняются параллельно с другими задачами из ROS и операционной системы, что вносит некоторую случайность в результаты. По этой причине в некоторых экспериментах мы приводим медианное значение для нескольких прогонов.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

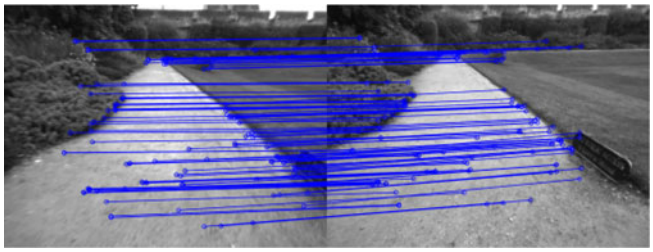

Рис. 4. Пример обнаружения петли в последовательности NewCollege. Мы проводим внутренние соответствия, подтверждающие найденное преобразование подобия.

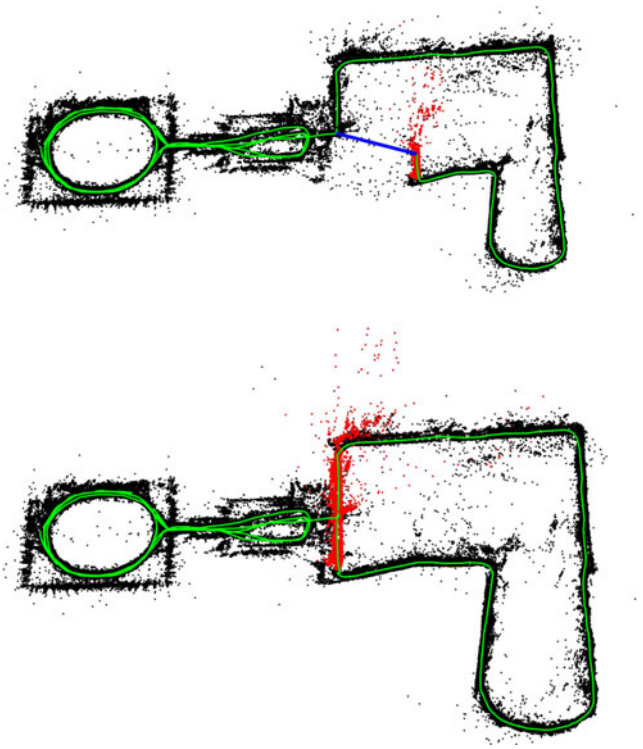

Рис. 5. Сопоставление до и после закрытия цикла в последовательности NewCollege. В совпадение замыкания контура отображается синим цветом, траектория - зеленым, а локальная карта для отслеживания в этот момент - красным. После закрытия локальная карта расширяется по обе стороны петли.

A. Производительность системы в наборе данных NewCollege

Набор данных NewCollege [39] содержит последовательность 2,2 км от робота, пересекающего кампус и прилегающие парки. Последовательность записывается стереокамерой со скоростью 20 кадров / с и разрешением512 × 382. Он содержит несколько петель и быстрых поворотов, что делает последовательность довольно сложной для монокулярного зрения. Насколько нам известно, в литературе нет другой монокулярной системы, способной обработать всю эту последовательность. Например Страсдат*и другие.* [7], несмотря на возможность замкнуть петли и работать в крупномасштабных средах, показал монокулярные результаты только для небольшой части этой последовательности.

В качестве примера нашей процедуры закрытия цикла на рис. 4 показано обнаружение цикла со вставками, которые поддерживают преобразование подобия. На рис. 5 показана реконструкция до и после замыкания петли. Красным цветом показана локальная карта, которая после замыкания петли распространяется по обеим сторонам петли.

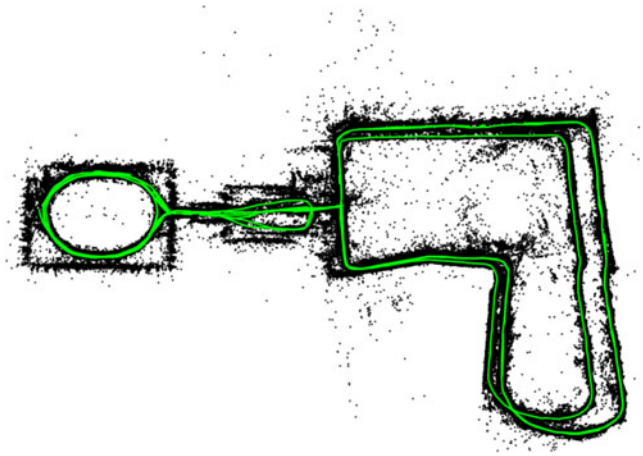

Рис. 6. ORB-SLAM реконструкция полной последовательности NewCollege. Большая петля справа пересечена в противоположных направлениях, и не было обнаружено замыканий визуальной петли; поэтому они не идеально совпадают.

ТАБЛИЦА I
ТСТОЙКА И МПРИЛОЖЕНИЕ TIMES IN NEWСОЛЛЕДЖ

Нить	Операция	Медиана (мс)	Среднее (мс)	Стандартное (мс)
ОТСЛЕЖИВАНИЕ	Извлечение ORB	11.10	11,42	1,61
	Начальная поза Расч.	3,38	3,45	0,99
	Отследить местную карту	14,84	16.01	9,98
	Всего	30,57	31,60	10,39
ЛОКАЛЬНАЯ КАРТА	Вставка ключевого кадра	10.29	11,88	5,03
	Отбор точек на карте	0,10	3,18	6,70
	Создание точки карты	66,79	72,96	31,48
	Местный БА	296,08	360,41	171,11
	Выбор ключевого кадра	8,07	15,79	18,98
	Всего	383,59	464,27	217,89

закрытие. Вся карта после обработки полной последовательности с ее реальной частотой кадров показана на рис. 6. Большой цикл справа не идеально выровнен, потому что он проходил в противоположных направлениях, и распознаватель места не смог найти замыкания цикла.

Мы извлекли статистику времени, затраченного каждым потоком в этом эксперименте. В таблице I показаны результаты отслеживания и локального сопоставления. Отслеживание работает с частотой кадров 25–30 Гц, что является наиболее сложной задачей для отслеживания локальной карты. При необходимости это время можно уменьшить, ограничив количество ключевых кадров, включаемых в локальную карту. В потоке локального сопоставления наиболее сложной задачей является локальный BA. В*местный БА* время меняется, если робот исследует или находится в хорошо нанесенной на карту области, потому что во время исследования BA прерывается, если отслеживание вставляет новый ключевой кадр, как объяснено в Разделе VE. В случае, если новые ключевые кадры не нужны, локальный BA выполняет большое количество итераций с префиксом.

В таблице II показаны результаты для каждого из шести найденных замыканий контура. Можно увидеть, как обнаружение петли сублинейно увеличивается с количеством ключевых кадров. Это связано с эффективным запросом к базе данных, которая сравнивает только подмножество изображений с общими словами, что демонстрирует потенциал набора слов для распознавания места. Наш*Essential Graph* включает ребра, примерно в пять раз превышающее количество ключевых кадров, что является довольно разреженным графом.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

ТАБЛИЦА II.
LOOП СПОТЕРЯ TIMES IN NEWСОЛЛЕДЖ

			Обнаружение петли (мс)		Коррекция петли		
Петля	Ключевые кадры	Основные грани графа	Выявление кандидатов	Преобразование подобия	Слияние	Основная оптимизация графов	Итого
1	287	1347	4,71	20,77	0,20	0,26	0,51
2	1082	5950	4,14	17,98	0,39	1.06	1,52
3	1279	7128	9,82	31,29	0,95	1,26	2,27
4	2648	12547	12,37	30,36	0,97	2.30	3,33
5	3150	16033	14,71	41,28	1,73	2,80	4,60
6	4496	21797	13,52	48,68	0,97	3,62	4,69

Б. Точность локализации в тесте TUM RGB-D

Тест TUM RGB-D [38] представляет собой отличный набор данных для оценки точности локализации камеры, так как он предоставляет несколько последовательностей с точными наземными данными, полученными с помощью внешней системы захвата движения. Мы отказались от всех тех последовательностей, которые, по нашему мнению, не подходят для чисто монокулярных систем SLAM, поскольку они содержат сильные вращения, отсутствие текстуры или движение.

Для сравнения мы также выполнили новые, прямые, полуидентичные LSD-SLAM [10] и PTAM [4] в тесте. Мы также сравниваем с траекториями, генерируемыми RGBD-SLAM [43], которые представлены для некоторых последовательностей на веб-сайте тестов. Чтобы сравнить ORB-SLAM, LSD-SLAM и PTAM с наземной истиной, мы выравниваем траектории ключевых кадров, используя преобразование подобия, поскольку масштаб неизвестен, и измеряем абсолютную ошибку траектории [38]. В случае RGBD-SLAM мы выравниваем траектории с помощью преобразования твердого тела, а также подобия, чтобы проверить, хорошо ли восстановлен масштаб. LSD-SLAM инициализируется из случайных значений глубины, и требуется время, чтобы сойтись; следовательно, мы отбросили первые десять ключевых кадров при сравнении с истинным значением. Для PTAM мы вручную выбрали два кадра, из которых мы получили хорошую инициализацию.

Видно, что ORB-SLAM способен обрабатывать все последовательности, кроме *fr3_nostructure_texture_far* (fr3_nstr_tex_ far). Это планарная сцена, которая, поскольку траектория камеры относительно плоскости имеет две возможные интерпретации, то есть двоякую неоднозначность, описанную в [27]. Наш метод инициализации обнаруживает двусмысленность и в целях безопасности отказывается от инициализации. PTAM инициализируется, выбирая иногда истинное решение, а иногда - поврежденное, и в этом случае ошибка недопустима. Мы не заметили двух разных реконструкций из LSD-SLAM, но ошибка в этой последовательности очень велика. В остальных последовательностях PTAM и LSD-SLAM демонстрируют меньшую надежность, чем наш метод, теряя трек в восьми и трех последовательностях соответственно.

С точки зрения точности ORB-SLAM и PTAM аналогичны в открытых траекториях, в то время как ORB-SLAM обеспечивает более высокую точность при обнаружении больших петель, как в последовательности *fr3_структура_текстура_около_с петлей* (fr3_nstr_tex_near). Самый удивительный результат заключается в том, что и PTAM, и ORB-SLAM явно более точны, чем LSD-SLAM и RGBD-SLAM. Одна из возможных причин может заключаться в том, что они сокращают оптимизацию карты до оптимизации позыграфа, когда измерения датчиков отбрасываются,

ТАБЛИЦА III
КРАМКА LOКАЛИЗАЦИЯ ERROR СОМПАРИЗОН В ТУМ RGB-D BENCHMARK [38]

	Абсолютная среднеквадратичная траектория ключевого кадра (см)			
	ORB-SLAM	ПТАМ	LSD-SLAM	RGBD- SLAM
fr1_xyz	**0,90**	1,15	9.00	1,34 (1,34)
fr2_xyz	0,30	**0,20**	2,15	2,61 (1,42)
fr1_floor	**2,99**	Икс	38,07	3,51 (3,51)
fr1_desk	**1,69**	Икс	10,65	2,58 (2,52)
fr2_360 _kidnap	3,81	**2,63**	Икс	393,3 (100,5)
fr2_desk	**0,88**	Икс	4,57	9,50 (3,94)
fr3_long _office	**3,45**	Икс	38,53	-
fr3_nstr_tex_far	обнаружена двусмысленность	4,92 / 34,74	18.31	-
fr3_nstr_ tex_near	**1,39**	2,74	7,54	-
fr3_str_tex_far	**0,77**	0,93	7,95	-
fr3_str_ tex_near	1,58	**1.04**	Икс	-
fr2_desk_person	**0,63**	Икс	31,73	6,97 (2,00)
fr3_sit_xyz	**0,79**	0,83	7,73	-
fr3_sit_halfsph	**1,34**	Икс	5,87	-
fr3_walk_xyz	**1,24**	Икс	12,44	-
fr3_walk_halfsph	**1,74**	Икс	Икс	-

Результаты для ORB-SLAM, PTAM и LSD-SLAM - это медиана для пяти выполнений в каждой последовательности. Траектории выровнены по 7 степеням свободы с наземной истиной. Траектории для RGBD-SLAM взяты с веб-сайта тестов, доступны только для последовательностей fr1 и fr2, и выровнены с 6 степенями свободы и 7 степенями свободы (результаты в скобках). X означает, что отслеживание в какой-то момент потеряно, и значительная часть последовательности не обрабатывается системой.

в то время как мы выполняем BA и совместно оптимизируем камеры и сопоставление с измерениями датчиков, что является золотым стандартом алгоритма для определения структуры по движению [2]. Мы обсудим этот результат более подробно в разделе IX-B. Другой интересный результат заключается в том, что LSD-SLAM кажется менее устойчивым к динамическим объектам, чем наша система, как показано на *fr2_desk_with_person* и *fr3_walking_xyz*.

Мы заметили, что RGBD-SLAM имеет смещение шкалы в *fr2* последовательности, так как выравнивание траекторий с 7 степенями свободы значительно снижает ошибку. Наконец, следует отметить, что Энгель *и другие.* [10] сообщил, что PTAM имеет меньшую точность, чем LSD-SLAM в *fr2_xyz* со среднеквадратичным отклонением 24,28 см. Однако в статье недостаточно подробностей о том, как были получены эти результаты, и мы не смогли их воспроизвести.

С. Перемещение в тесте TUM RGB-D Benchmark

Мы проводим два эксперимента по перемещению в тесте TUM RGB-D. В первом эксперименте мы строим карту с первыми 30 секундами последовательности *fr2_xyz* и выполнять глобальное перемещение с каждым последующим кадром и оценивать точность

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

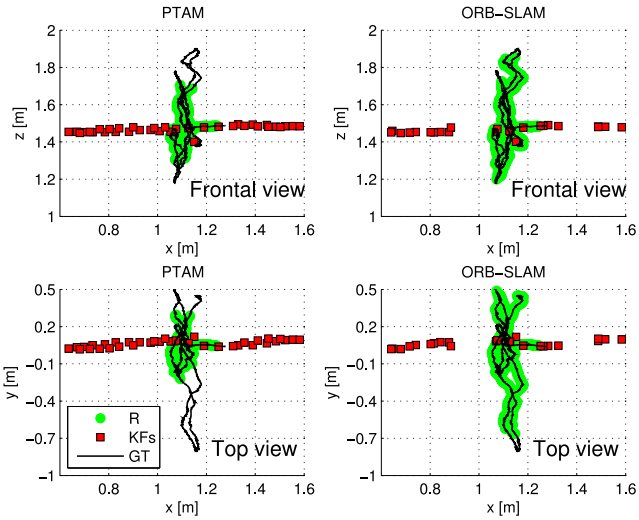

Рис. 7. Эксперимент по релокализации в *fr2_xyz*. Карта изначально создается в течение первых 30 секунд последовательности (KF). Цель состоит в том, чтобы переместить последующие кадры. Показаны успешные перемещения (R) нашей системы и PTAM. Наземная истина (GT) отображается только для кадров, которые нужно переместить.

Рис. 8. Пример сложных перемещений (резкое изменение масштаба, динамические объекты), которые наша система успешно обнаружила в экспериментах по перемещению.

ТАБЛИЦА IV
РЕЗУЛЬТАТЫ ДЛЯ РЭЛОКАЛИЗАЦИЯ ЭКСПЕРИМЕНТЫ

	Начальная карта		Перебазирование		
Система	KFs	RMSE (см)	Отзывать (%)	RMSE (см)	Максимум. Ошибка (см)
fr2_xyz. 2769 кадров для перемещения					
PTAM	37	0,19	34,9	0,26	1,52
ORB-SLAM	24	0,19	**78,4**	0,38	1,67
fr3_walking_xyz. 859 кадров для перемещения					
PTAM	34	0,83	0,0	-	-
ORB-SLAM	31 год	0,82	**77,9**	1,32	4,95

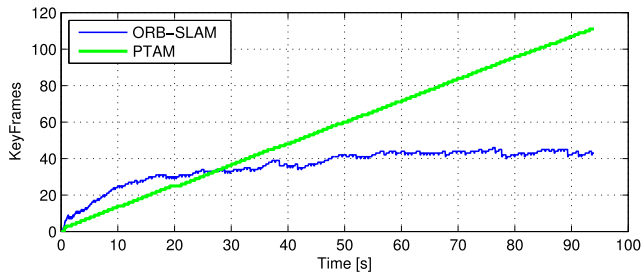

Рис.9. Пожизненный эксперимент в статической среде, где камера всегда смотреть на одно и то же место с разных точек зрения. PTAM всегда вставляет ключевые кадры, в то время как ORB-SLAM может удалять избыточные ключевые кадры и поддерживать карту ограниченного размера.

восстановленных поз. Мы проводим тот же эксперимент с PTAM для сравнения. На рис. 7 показаны ключевые кадры, использованные для создания исходной карты, позы перемещенных кадров и исходная информация для этих кадров. Можно видеть, что PTAM может перемещать только кадры, которые находятся рядом с ключевыми кадрами из-за небольшой неизменности его метода перемещения. Таблица IV показывает отзыв и ошибку по отношению к истине. ORB-SLAM точно перемещает более двух кадров, чем PTAM. Во втором эксперименте мы создаем начальную карту с последовательностью *fr3_sitting_xyz* и попробуйте переместить все кадры из *fr3_walking_xyz*. Это сложный эксперимент, так как есть большие окклюзии из-за движущихся по сцене людей. Здесь PTAM не находит перемещений, в то время как наша система перемещает 78% фреймов, как можно увидеть в Таблице IV. На рис. 8 показаны некоторые примеры сложных перемещений, выполненных нашей системой в этих экспериментах.

D. Пожизненный эксперимент в тесте TUM RGB-D

Предыдущие эксперименты по перемещению показали, что наша система способна локализоваться на карте с самых разных точек обзора и надежно при умеренных динамических изменениях. Это свойство в сочетании с нашей процедурой отбора ключевых кадров позволяет нам работать в течение всей жизни в одной и той же среде под разными точками зрения и некоторыми динамическими изменениями.

В случае полностью статического сценария наша система может поддерживать количество ограниченных ключевых кадров, даже если камера смотрит на сцену с разных точек обзора. Мы демонстрируем это в настраиваемой последовательности, где камера смотрит на один и тот же стол в течение 93 секунд, но выполняет траекторию, так что точка обзора всегда меняется. Мы сравниваем эволюцию количества ключевых кадров на нашей карте и тех, которые генерируются PTAM на рис. 9. Можно увидеть, как PTAM всегда вставляет ключевые кадры, в то время как наш механизм сокращения избыточных ключевых кадров приводит к насыщению их числа.

Хотя непрерывная работа в статическом сценарии должна быть требованием любой системы SLAM, более интересным является случай, когда происходят динамические изменения. Мы анализируем поведение нашей системы в таком сценарии, последовательно выполняя динамические последовательности из*fr3*: *сидит_xyz*, *сидит_полусфера*, *sit_rpy*, *walking_xyz*, *walking_halfspehere*, и *walking_rpy*. Во всех эпизодах камера фокусируется на одном столе, но движется по разным траекториям, в то время как люди движутся и меняют некоторые объекты, например стулья. Рис. 10 (a) показывает эволюцию

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

общее количество ключевых кадров на карте, а на рис. 10 (b) для каждого ключевого кадра показан его кадр создания и уничтожения, показывающий, как долго ключевые кадры сохраняются на карте. Видно, что во время первых двух последовательностей размер карты увеличивается, поскольку все виды сцены видны впервые. На рис. 10 (b) мы видим, что несколько ключевых кадров, созданных во время этих двух первых последовательностей, сохраняются на карте в течение всего эксперимента. Во время последовательностей *sit_rpy* и *walking_xyz*, карта не увеличивается, потому что созданная карта хорошо объясняет сцену. Напротив, во время последних двух последовательностей вставляется больше ключевых кадров, показывая, что в сцене есть некоторые новинки, которые еще не были представлены, вероятно, из-за динамических изменений. Наконец, на рис. 10 (c) показана гистограмма ключевых кадров в зависимости от времени, в течение которого они выжили, по отношению к оставшемуся времени последовательности с момента ее создания. Можно видеть, что большинство ключевых кадров уничтожаются процедурой отбраковки вскоре после создания, и только небольшое подмножество доживает до конца эксперимента. С одной стороны, это показывает, что наша система имеет обширную политику создания ключевых кадров, которая очень полезна при выполнении резких движений при исследовании. С другой стороны,

В этих экспериментах на протяжении всей жизни мы показали, что наша карта растет вместе с содержимым сцены, но не со временем, и способна сохранять динамические изменения сцены, что может быть полезно для понимания сцены путем накопления опыта в окружающей среде. .

Д. Крупномасштабное и закрытие большого цикла в наборе данных KITTI

Тест одометрии из набора данных KITTI [40] содержит 11 последовательностей, снятых автомобилем, проезжающим по жилому району, с точными данными GPS и лазерного сканера Velodyne. Это очень сложный набор данных для монокулярного зрения из-за быстрого вращения, участков с большим количеством листвы, которые затрудняют сопоставление данных, и относительно высокой скорости автомобиля, поскольку последовательности записываются со скоростью 10 кадров / с. Мы воспроизводим последовательности с реальной частотой кадров, с которой они были записаны, и ORB-SLAM может обрабатывать все последовательности, за исключением последовательности 01, которая представляет собой шоссе с несколькими отслеживаемыми близкими объектами. Последовательности 00, 02, 05, 06, 07 и 09 содержат циклы, которые были правильно обнаружены и закрыты нашей системой. Последовательность 09 содержит цикл, который можно обнаружить только в нескольких кадрах в конце последовательности,

Качественные сравнения наших траекторий и истины показаны на рис. 11 и 12. Как и в тесте TUM RGB-D, мы выровняли траектории ключевых кадров нашей системы и основную истину с помощью преобразования подобия. Мы можем качественно сравнить наши результаты с рис. 11 и 12 с результатами, полученными для последовательностей 00, 05, 06, 07 и 08 с помощью недавнего монокулярного подхода SLAM Лим *и другие*. [25, рис. 10]. ORB-SLAM производит явно более точные траектории для всех этих последовательностей, за исключением последовательности 08, в которой они, кажется, менее подвержены дрейфу.

В таблице V показана средняя среднеквадратичная ошибка траектории ключевого кадра за пять выполнений в каждой последовательности. Мы также предоставляем

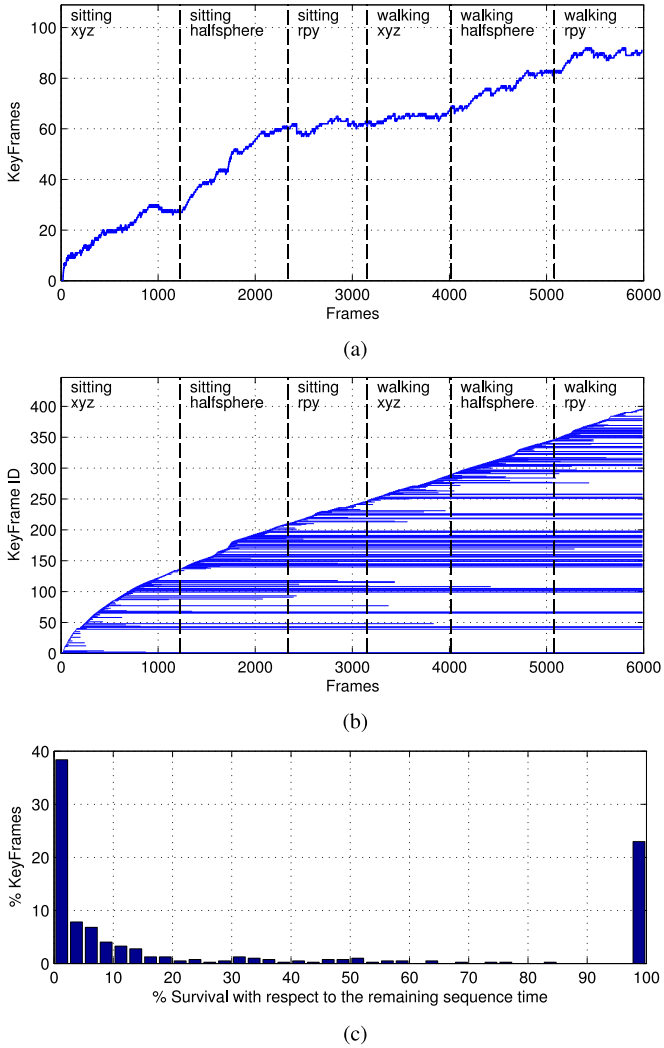

Рис 10. Пожизненный эксперимент в динамической среде от TUM Тест RGB-D. (a) Эволюция количества ключевых кадров на карте. (б) Создание и уничтожение ключевых кадров. Каждая горизонтальная линия соответствует ключевому кадру от кадра создания до его разрушения. (c) Гистограмма времени выживания всех порожденных ключевых кадров по отношению к оставшемуся времени эксперимента.

размеры карт, чтобы поместить в контекст ошибки. Результаты демонстрируют, что наша система очень точна, поскольку ошибка траектории обычно составляет около 1% от ее размеров, иногда меньше, как в последовательности 03 с ошибкой 0,3% или выше, как в последовательности 08 с 5%. В последовательности 08 петли отсутствуют, и дрейф нельзя исправить, что делает очевидной необходимость замыкания петель для достижения точных реконструкций.

В этом эксперименте мы также проверили, насколько можно улучшить реконструкцию, выполнив 20 итераций *полный бакалавр* (подробности см. в Приложении) в конце каждой последовательности. Мы заметили, что некоторые итерации *полный бакалавр* немного улучшает точность траекторий с петлями, но оказывает незначительное влияние на открытые траектории, что означает, что выходные данные нашей системы уже очень точны. В любом случае, если требуются наиболее точные результаты, наш алгоритм предоставляет набор совпадений, которые определяют надежную сеть камер, и первоначальное предположение, так что *полный бакалавр* сходятся за несколько итераций.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

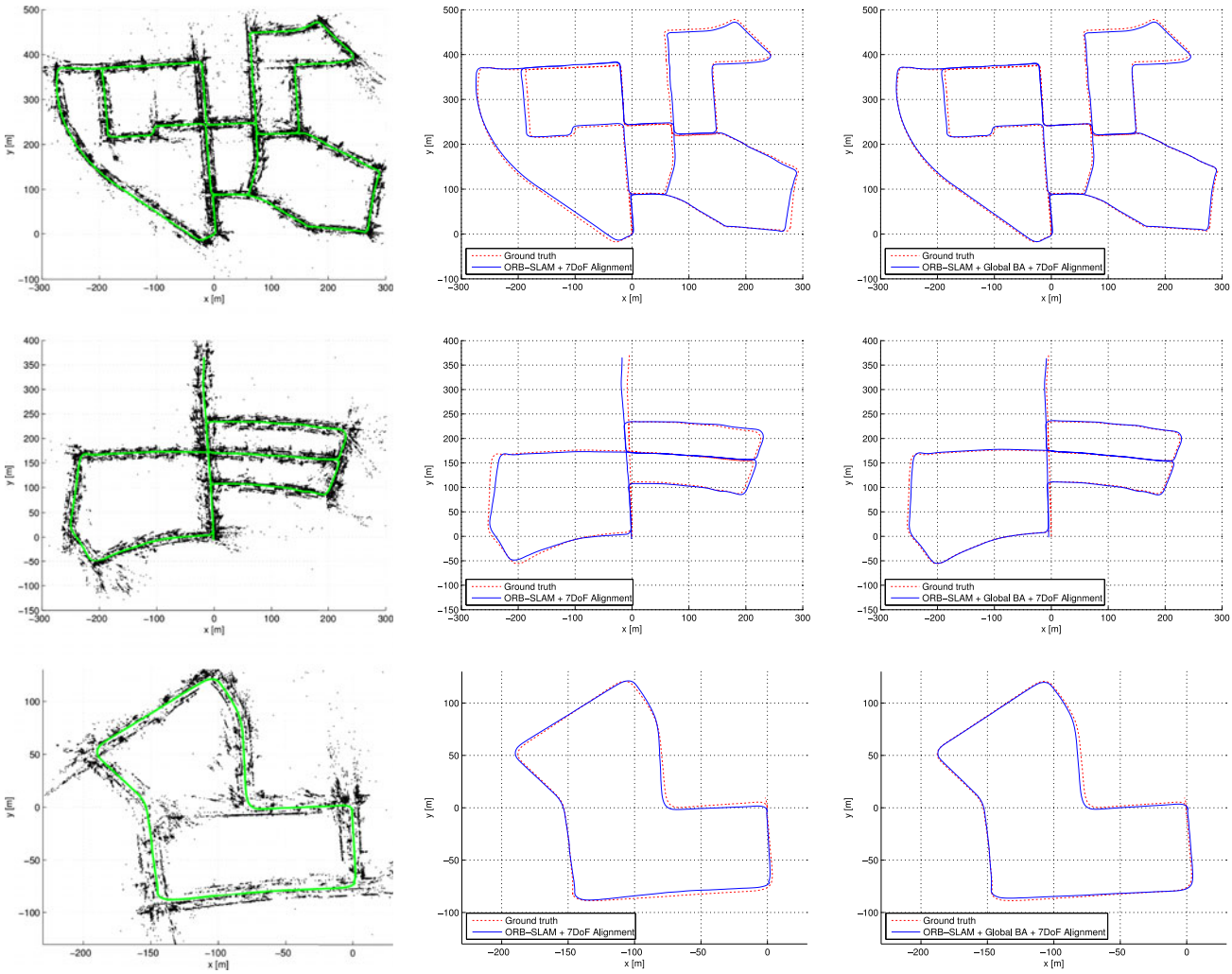

Рис. 11. Последовательности 00, 05 и 07 из эталонного теста одометрии набора данных KITTI. (Слева) Точки и траектория ключевого кадра. (В центре) траектория и наземная правда. (Справа) Траектория после 20 итераций полного БА. Вывод нашей системы довольно точен, хотя его можно немного улучшить с помощью некоторых итераций BA.

Наконец, мы хотели показать эффективность нашего подхода к замыканию цикла и влияние $\theta_{\text{мин}}$ используется для включения ребер в существенный граф. Мы выбрали последовательность 09 (очень длинную последовательность с закрытием цикла в конце), и в одном и том же исполнении мы оценили разные стратегии закрытия цикла. В Таблице VI мы показываем среднеквадратичную траекторию ключевого кадра и время, затраченное на оптимизацию в различных случаях: без закрытия цикла, если мы непосредственно применяем *полный бакалавр* (20 или 100 итераций), если мы применим только оптимизацию графа позы (десять итераций с разным количеством ребер), и если мы применим оптимизацию графа позы и *полный бакалавр* после. На рис. 13 показаны выходные траектории различных методов. Результаты ясно показывают, что до закрытия цикла решение настолько далеко от оптимального, что у BA возникают проблемы сходимости. Даже после 100 итераций ошибка все равно очень высока. С другой стороны, существенная оптимизация графа показывает быструю сходимость и более точные результаты. Видно, что выбор $\theta_{\text{мин}}$ не оказывает существенного влияния на точность, но, уменьшив количество ребер, время можно значительно сократить. Выполнение дополнительной BA после оптимизации графа позы немного повышает точность, при этом существенно увеличивая время.

IX. Свключение и Добсуждение

A. Выводы

В этом исследовании мы представили новую монокулярную систему SLAM с подробным описанием ее строительных блоков и исчерпывающей оценкой в общедоступных наборах данных. Наша система продемонстрировала, что она может обрабатывать последовательности сцен в помещении и на улице, а также движения автомобиля, робота и рук. Точность системы обычно составляет менее 1 см в небольших помещениях и несколько метров в больших сценариях на открытом воздухе (после того, как мы выровняли шкалу с истинностью на земле).

В настоящее время PTAM Клейна и Мюррея [4] считается наиболее точным методом SLAM для монокулярного видео в реальном времени. Неслучайно внутренняя часть PTAM - это BA, который, как хорошо известно, является золотым стандартом для решения автономной проблемы структуры из движения [2]. Одним из главных успехов PTAM и более ранней работы Mouragnon [3] было привнесение этих знаний в сообщество SLAM робототехники и демонстрация их производительности в реальном времени. Главный вклад нашей работы состоит в расширении универсальности PTAM в средах, которые не поддаются обработке для этой системы. Для этого мы разработали

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

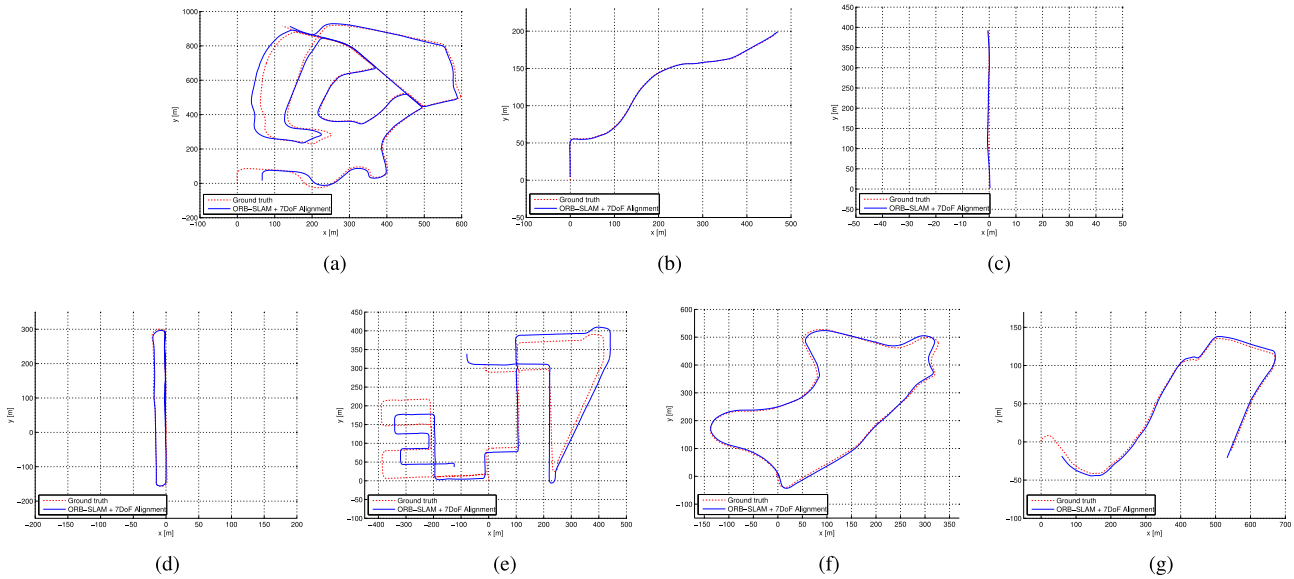

Рис. 12. Траектории ключевых кадров ORB-SLAM в последовательностях 02, 03, 04, 06, 08, 09 и 10 из теста одометрии набора данных KITTI. Последовательность 08 не содержит петель и дрейф (особенно масштаб) не исправляется. (a) Последовательность 02. (b) Последовательность 03. (c) Последовательность 04. (d) Последовательность 06. (e) Последовательность 08 (f) Последовательность 09. (g) Последовательность 10.

с нуля новая монокулярная система SLAM с некоторыми новыми идеями и алгоритмами, но также включающая в себя отличные работы, разработанные за последние несколько лет, такие как обнаружение петель Гальвеса-Лопеса и Тардоса [5], процедура замыкания петли и граф ковидимости Страсдата *и другие*. [6], [7], фреймворк оптимизации g2o Куэммерле *и другие*. [37], и функции ORB от Rubble *и другие*. [9]. Насколько нам известно, ни одна другая система не продемонстрировала свою работоспособность в таком количестве различных сценариев и с такой точностью. Поэтому наша система на данный момент является наиболее надежным и полным решением для монокулярного SLAM. Наша новая политика создания и отбраковки ключевых кадров позволяет создавать ключевые кадры каждые несколько кадров, которые в конечном итоге удаляются, если они считаются избыточными. Это гибкое расширение карты действительно полезно в плохо обусловленных исследовательских траекториях, то есть близких к чистым поворотам или быстрым движениям. При многократной работе в одной и той же среде карта увеличивается только в том случае, если визуальное содержимое сцены изменяется, сохраняя историю ее различных визуальных проявлений. Интересные результаты для долгосрочного картирования можно получить, анализируя эту историю.

Наконец, мы также продемонстрировали, что функции ORB обладают достаточной способностью распознавания, чтобы обеспечить распознавание места при серьезном изменении точки зрения. Более того, они настолько быстро извлекаются и сопоставляются (без необходимости многопоточности или ускорения графического процессора), что обеспечивает точное отслеживание и отображение в реальном времени.

Б. Разреженные / основанные на признаках методы против плотных / прямых методов

Недавние алгоритмы SLAM для монокуляров в реальном времени, такие как DTAM [44] и LSD-SLAM [10], могут выполнять плотные или полуидентичные реконструкции окружающей среды, в то время как камера локализуется путем оптимизации непосредственно по интенсивности пикселей изображения. Эти прямые подходы не требуют извлечения признаков и, таким образом, позволяют избежать соответствующих артефактов. Они также более устойчивы к размытию, малотекстурной среде и

ТАБЛИЦА V
РЕЗУЛЬТАТЫ НАШЕЙ СИСТЕМЫ В НАБОРЕ ДАННЫХ KITTI

		ORB-SLAM		+ Global BA (20 шт.)	
Последовательность	Размер (м×м)	KFs	RMSE (м)	RMSE (м)	Время BA (с)
КИТТИ 00	564 × 496	1391	6,68	5,33	24,83
КИТТИ 01	1157 × 1827 г.	Икс	Икс	Икс	Икс
КИТТИ 02	599 × 946	1801 г.	21,75	21,28	30.07
КИТТИ 03	471 × 199	250	1,59	1,51	4.88
КИТТИ 04	0.5 × 394	108	1,79	1,62	1,58
КИТТИ 05	479 × 426	820	8,23	4,85	15.20
КИТТИ 06	23 × 457	373	14,68	12,34	7,78
КИТТИ 07	191 × 209	351	3,36	2,26	6,28
KITTI 08	808 × 391	1473	46,58	46,68	25,60
КИТТИ 09	465 × 568	653	7,62	6,62	11,33
КИТТИ 10	671 × 177	411	8,68	8,80	7,64

высокочастотная текстура наподобие асфальта [45]. Их более плотные реконструкции по сравнению с разреженной точечной картой нашей системы или PTAM могут быть более полезными для других задач, чем просто локализация камеры.

Однако у прямых методов есть свои ограничения. Во-первых, эти методы предполагают модель отражательной способности поверхности, которая в реальных сценах создает свои собственные артефакты. Фотометрическая согласованность ограничивает базовую линию совпадений, обычно более узкую, чем позволяют характеристики. Это имеет большое влияние на точность реконструкции, которая требует обширных базовых наблюдений для уменьшения неопределенности глубины. Прямые методы, если они не смоделированы правильно, в значительной степени подвержены влиянию артефактов скользящего затвора, автоусиления и автоэкспозиции (как в тесте TUM RGB-D Benchmark). Наконец, поскольку прямые методы, как правило, очень требовательны к вычислениям, карта просто постепенно расширяется, как в DTAM, или оптимизация карты сводится к графу поз, отбрасывая все измерения датчиков, как в LSD-SLAM.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

ТАБЛИЦА VI
СОМПАРИЗОН LOOП СПОТЕРЯ STРАТЕГИИ В КИТТИ 09

Метод	Время (с)	Края графа позы	RMSE (м)
-	-	-	48,77
BA (20)	14,64	-	49,90
BA (100)	72,16	-	18,82
Например (200)	0,38	890	8,84
Например (100)	0,48	1979 г.	8,36
Например (50)	0,59	3583	8,95
Например (15)	0,94	6663	8,88
EG (100) + BA (20)	13,40	1979 г.	7,22

В первой строке показаны результаты без закрытия цикла. Число в скобках для BA означает количество итераций Левенберга – Марквардта (LM), в то время как для EG (существенный граф) оно равно$\theta_{мин}$ построить существенный граф. Все оптимизации EG выполняют десять итераций LM.

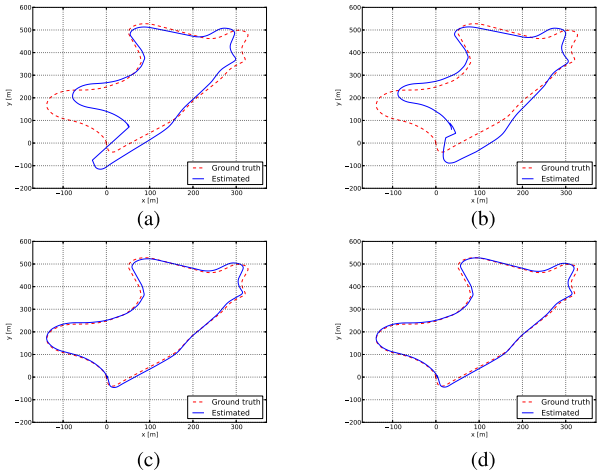

Рис 13. Сравнение различных стратегий закрытия петель в KITTI 09. (а) Без замыкания петли. (б) БА (20). (с) EG (100). (г) ЭГ (100) + БА (20).

Напротив, методы, основанные на признаках, могут сопоставлять объекты с широкой базовой линией благодаря их хорошей инвариантности к изменениям точки обзора и освещения. BA совместно оптимизирует позы камеры и указывает на измерения сенсора. В контексте оценки структуры и движения Торр и Зиссерман [46] уже указали на преимущества методов, основанных на признаках, по сравнению с прямыми методами. В этом исследовании мы предоставляем экспериментальные доказательства (см. Раздел VIII-B) превосходной точности основанных на функциях методов в SLAM в реальном времени. Мы считаем, что будущее монокулярного SLAM должно включать лучшее из обоих подходов.

C. Будущая работа

Точность нашей системы еще можно улучшить, добавив в отслеживание бесконечно удаленных точек. Эти точки, которые не видны с достаточным параллаксом и которые наша система не включает в карту, очень информативны для вращения камеры [21].

Другой открытый способ - обновить разреженную карту нашей системы до более плотной и полезной реконструкции. Благодаря нашему выбору ключевых кадров, ключевые кадры представляют собой компактную сводку окружающей среды с очень высокой точностью позы и богатой информацией о видимости. Следовательно, разреженная карта ORB-SLAM может быть отличным исходным предположением и скелетом, помимо что позволяет построить плотную и точную карту сцены. Первые попытки в этом направлении представлены в [47].

АПРИЛОЖЕНИЕ
NОНЛАЙН ОПТИМИЗАЦИИ

1) *Регулировка связки [1]:* Расположение точек на карте 3-D $\mathbf{Икс}_{ш,\,дж} \in \mathrm{p}3$ и позы ключевых кадров $\mathbf{T}_{iw} \in \mathrm{SE}(3)$, где $ш$ обозначает мировой справочник, оптимизированы для минимизации ошибки перепроецирования по отношению к совпадающим ключевым точкам $\mathbf{Икс}_{я,\,j} \in \mathrm{p}2$. Срок погрешности при наблюдении за точкой на карте j в ключевом кадре $я$ является

$$\mathbf{e}_{я,\,j} \text{ знак равно } \mathbf{Икс}_{я,\,j} - \pi_{я}(\mathbf{T}_{iw}, \mathbf{Икс}_{ш,\,дж}) \tag{5}$$

где $\pi_{я}$ функция проекции

$$\pi_{я}(\mathbf{T}_{iw}, \mathbf{Икс}_{ш,\,дж} \text{ знак равно } \begin{bmatrix} \dfrac{Икс_{я,\,j}}{z_{я,\,j}} + c_{я,ты} \\ \dfrac{y_{я,\,j}}{z_{я,\,j}} + c_{я,\,v} \end{bmatrix}$$

$$\begin{bmatrix} Икс_{я,\,j} & y_{я,\,j} & z_{я,\,j} \end{bmatrix}^{T} \text{ знак равно } \mathbf{p}_{iw}\mathbf{Икс}_{ш,\,дж} + \mathbf{t}_{iw} \tag{6}$$

где $\mathbf{p}_{iw} \in \mathrm{TAK}(3)$ и $\mathbf{t}_{iw} \in \mathrm{p}3$ - соответственно вращательная и трансляционная части $\mathbf{T}_{iw}$, и $(f_{я,\,ты}, f_{я,\,v})$ и $(c_{я,\,ты}, c_{я,\,v})$ фокусное расстояние и основная точка камеры $я$. Минимизируемая функция стоимости:

$$C \text{ знак равно } \sum_{я,\,j} \rho_{час}(\mathbf{e}_{я,\,j}^{T}\boldsymbol{\Omega}_{я,\,j}\mathbf{e}_{я,\,j}) \tag{7}$$

где $\rho_{час}$ - робастная функция стоимости Хубера, и $\boldsymbol{\Omega}_{я,\,j}$ знак равно $\sigma_{я,\,j}^{-2}\mathbf{I}_{2\times 2}$ ковариационная матрица, связанная с масштабом в котором была обнаружена ключевая точка. В случае *полный бакалавр* (используется при инициализации карты, описанной в Разделе IV, и в экспериментах в Разделе VIII-E), мы оптимизируем все точки и ключевые кадры, за исключением первого ключевого кадра, который остается фиксированным в качестве источника. В *местный БА* (см. Раздел VI-D), все точки, включенные в локальную область, оптимизированы, в то время как подмножество ключевых кадров фиксировано. При оптимизации позы, или *только движение BA* (см. Раздел V), все точки фиксированы, и оптимизирована только поза камеры.

2) *Оптимизация поз-графа над Sim (3) Ограничения [6]:* Учитывая граф поз двоичных ребер (см. Раздел VII-D), мы определяем ошибку на ребре как

$$\mathbf{e}_{я,\,j} \text{ знак равно бревно}_{\mathrm{Sim}(3)}(\mathbf{S}_{ij}\mathbf{S}_{jw}\mathbf{S}_{iw}^{-1}) \tag{8}$$

где $\mathbf{S}_{ij}$ - относительное преобразование Sim (3) между обоими ключевыми кадрами, вычисленное из поз SE (3) непосредственно перед оптимизацией графа позы и установкой масштабного коэффициента равным 1. В случае края замыкания цикла это относительное преобразование вычисляется с помощью метода Хорна [42]. Вбревно$_{\mathrm{Sim}3}$ [48] преобразуется в касательное пространство, так что ошибка является вектором в $\mathrm{p}7$. Цель состоит в том, чтобы оптимизировать ключевые кадры Sim (3), минимизируя функцию стоимости как

$$C \text{ знак равно } \sum_{я,\,j} (\mathbf{e}_{я,\,j}^{T}\boldsymbol{\Lambda}_{я,\,j}\mathbf{e}_{я,\,j}) \tag{9}$$

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

где $\mathbf{\Lambda}_{я,\,j}$ - информационная матрица ребра, которую, как и в [48], мы устанавливаем равной единице. Мы фиксируем ключевой кадр замыкания петли, чтобы зафиксировать 7 степеней свободы шкалы. Хотя этот метод является грубым приближением *полный бакалавр*, мы экспериментально демонстрируем в разделе VIII-E, что он имеет значительно более быструю и лучшую сходимость, чем BA.

3) *Relative Sim (3) Оптимизация:* Учитывая набор $п$ Спички $я \Rightarrow j$ (ключевые точки и связанные с ними точки трехмерной карты) между ключевыми кадрами 1 и ключевой кадр 2, мы хотим оптимизировать относительное преобразование Sim (3) $\mathbf{S}_{12}$ (см. Раздел VII-B), который сводит к минимуму ошибку перепроецирования в обоих изображениях как

$$\mathbf{e}_1 \text{ знак равно } \mathbf{Икс}_{1,я} - \pi_1(\mathbf{S}_{12}, \mathbf{Икс}_{2,\,j})$$
$$\mathbf{e}_2 \text{ знак равно } \mathbf{Икс}_{2,\,j} - \pi_2(\mathbf{S}_{12}, \mathbf{Икс}_{1,я}) \tag{10}$$

а функция стоимости, которую необходимо минимизировать, равна

$$\sum_{C \text{ знак равно}} \left(\rho_{час}(\mathbf{e}_1^T \mathbf{\Omega}_{1,я}^{-1} \mathbf{e}_1) + \rho_{час}(\mathbf{e}_2^T \mathbf{\Omega}_{2,\,j}^{-1} \mathbf{e}_2) \right) \tag{11}$$
$$п$$

где $\mathbf{\Omega}_{1,я}$ и $\mathbf{\Omega}_{2,я}$ - это ковариационные матрицы, связанные со шкалой, в которой были обнаружены ключевые точки на изображениях 1 и 2. В этой оптимизации точки фиксированы.

рЭФФЕРЕНЦИИ

[1] Б. Триггс, П. Ф. Маклаучлан, Р. И. Хартли и А. В. Фитцгиббон, «Связанное согласование - современный синтез», в *Алгоритмы зрения: теория и практика* . Нью-Йорк, штат Нью-Йорк, США: Springer, 2000, стр. 298–372.

[2] Р. Хартли и А. Зиссерман, *Многоканальная геометрия в компьютерном зрении*, 2-е изд., Кембридж, Великобритания: Cambridge Univ. Пресса, 2004.

[3] Э. Мураньон, М. Луйер, М. Дом, Ф. Декейзер и П. Сайд, «Локализация в реальном времени и трехмерная реконструкция», в *Proc. IEEE Comput. Soc. Конф. Comput. Vision Pattern Recog.*, 2006, т. 1. С. 363–370.

[4] Г. Кляйн и Д. Мюррей, «Параллельное отслеживание и отображение для небольших рабочих пространств AR», в *Proc. IEEE ACM Int. Symp. Смешанная дополненная реальность*, Нара, Япония, ноябрь 2007 г., стр. 225–234.

[5] Д. Гальвес-Лопес и Дж. Д. Тардос, «Мешки двоичных слов для быстрого распознавания места в последовательностях изображений», *IEEE Trans. Робот.*, т. 28, вып. 5. С. 1188–1197, октябрь 2012 г.

[6] Х. Страсдат, Дж. М. М. Монтиель и А. Дж. Дэвисон, «Масштабный монокуляр SLAM с учетом дрейфа», представленный на Proc. Робот .: Науки. Syst., Сарагоса, Испания, июнь 2010 г.

[7] Х. Страсдат, Эй Джей Дэвисон, Дж. М. М. Монтиель и К. Конолиге, «Оптимизация двойного окна для визуального SLAM с постоянным временем», in *Proc. IEEE Int. Конф. Comput. Зрение*, Барселона, Испания, ноябрь 2011 г., стр. 2352–2359.

[8] К. Мей, Г. Сибли, П. Ньюман, «Замыкание петель без мест», in *Proc. IEEE / RSJ Int. Конф. Intell. Роботы Syst.*, Тайбэй, Тайвань, октябрь 2010 г., стр. 3738–3744.

[9] Э. Рубли, В. Рабо, К. Конолидже и Г. Брадски, «ORB: эффективная альтернатива SIFT или SURF», in *Proc. IEEE Int. Конф. Comput. Зрение*, Барселона, Испания, ноябрь 2011 г., стр. 2564–2571.

[10] Дж. Энгель, Т. Шёпс и Д. Кремерс, «LSD-SLAM: крупномасштабный прямой монокулярный SLAM», в *Proc. Евро. Конф. Comput. Зрение*, Цюрих, Швейцария, сентябрь 2014 г., стр. 834–849.

[11] Р. Мур-Арталь и Дж. Д. Тардос, «Быстрое перемещение и закрытие цикла в SLAM на основе ключевых кадров», в *Proc. IEEE Int. Конф. Робот. Автомат.*, Гонконг, июнь 2014 г., стр. 846–853.

[12] Р. Мур-Арталь и Дж. Д. Тардос, «ORB-SLAM: отслеживание и отображение распознаваемых элементов», представленные на роботе MVIGRO Workshop. Sci. Syst., Беркли, Калифорния, США, июль 2014 г.

[13] Б. Уильямс, М. Камминс, Дж. Нейра, П. Ньюман, И. Рид и Дж. Д. Тардос, «Сравнение методов замыкания петель в монокулярном SLAM», *Робот. Auton. Syst.*, т. 57, нет. 12. С. 1188–1197, 2009.

[14] Д. Нистер и Х. Стюениус, «Масштабируемое распознавание с помощью словарного дерева», в *Proc. IEEE Comput. Soc. Конф. Comput. Vision Pattern Recog.*, Нью-Йорк, Нью-Йорк, США, июнь 2006 г., т. 2. С. 2161–2168.

[15] М. Камминс и П. Ньюман, «SLAM только для внешнего вида в больших масштабах с FAB-MAP 2.0», *Int. J. Робот. Res.*, т. 30, нет. 9. С. 1100–1123, 2011.

[16] М. Калондер, В. Лепет, К. Стреча, П. Фуа, «КРАТКОЕ: бинарные робастные независимые элементарные функции», в *Proc. Евро. Конф. Comput. Зрение*, Херсониссос, Греция, сентябрь 2010 г., стр. 778–792.

[17] Э. Ростен и Т. Драммонд, «Машинное обучение для высокоскоростного обнаружения углов», в *Proc. Евро. Конф. Comput. Зрение*, Грац, Австрия, май 2006 г., стр. 430–443.

[18] Х. Бэй, Т. Туйтелаарс и Л. Ван Гул, «SURF: улучшенные надежные функции», в *Proc. Евро. Конф. Comput. Зрение*, Грац, Австрия, май 2006 г., стр. 404–417.

[19] Д.Г. Лоу, «Отличительные особенности изображения от масштабно-инвариантных ключевых точек», *Int. J. Comput. Зрение*, т. 60, нет. 2. С. 91–110, 2004.

[20] AJ Davison, ID Reid, ND Molton и O. Stasse, «MonoSLAM: SLAM с одной камерой в реальном времени», *IEEE Trans. Pattern Anal. Max. Intell.*, т. 29, нет. 6. С. 1052–1067, июнь 2007 г.

[21] Дж. Сивера, А. Дж. Дэвисон и Дж. М. М. Монтиель, «Параметризация обратной глубины для монокулярного SLAM», *IEEE Trans. Робот.*, т. 24, вып. 5, С. 932–945, октябрь 2008 г.

[22] К. Форстер, М. Пиццоли и Д. Скарамуцца, «SVO: Быстрая полупрямая монокулярная визуальная одометрия», in *Proc. IEEE Int. Конф. Робот. Автомат.*, Гонконг, июнь 2014 г., стр. 15–22.

[23] OD Faugeras, F. Lustman, "Движение и структура из движения в кусочно-плоской среде", *Int. J. Pattern Recog. Артиф. Intell.*, т. 2, вып. 03. С. 485–508, 1988.

[24] В. Тан, Х. Лю, З. Донг, Г. Чжан и Х. Бао, «Надежный монокулярный SLAM в динамических средах», в *Proc. IEEE Int. Symp. Смешанная дополненная реальность*, Аделаида, Австралия, октябрь 2013 г., стр. 209–218.

[25] Х. Лим, Дж. Лим и Х. Дж. Ким, «Монокулярный визуальный SLAM с 6 степенями свободы в реальном времени в крупномасштабной среде», в *Proc. IEEE Int. Конф. Робот. Автомат.*, Гонконг, июнь 2014 г., стр. 1532–1539.

[26] Д. Нистер, "Эффективное решение проблемы пяти точек относительной позы", *IEEE Trans. Pattern Anal. Max. Intell.*, т. 26, вып. 6. С. 756–770, июнь 2004 г.

[27] Х. Лонге-Хиггинс, «Реконструкция плоской поверхности из двух перспективных проекций», *Proc. Royal Soc. Лондон сер. B, Biol, Sci.*, т. 227, нет. 1249. С. 399–410, 1986.

[28] PH Torr, AW Fitzgibbon, A. Zisserman, "Проблема вырождения структуры и восстановления движения из некалиброванных последовательностей изображений", *Int. J. Comput. Зрение*, т. 32, нет. 1. С. 27–44, 1999.

[29] А. Кьюзо, П. Фаваро, Х. Джин и С. Соатто, «Структура движения, причинно интегрированная во времени», *IEEE Trans. Pattern Anal. Max. Intell.*, т. 24, вып. 4. С. 523–535, апрель 2002 г.

[30] Э. Ид и Т. Драммонд, «Масштабируемый монокуляр SLAM», в *Proc. IEEE Comput. Soc. Конф. Comput. Vision Pattern Recog.*, Нью-Йорк, Нью-Йорк, США, июнь 2006 г., т. 1. С. 469–476.

[31] Х. Страсдат, Дж. М. М. Монтиель и А. Дж. Дэвисон, «Визуальный SLAM: зачем фильтровать?» *Image Vision Comput.*, т. 30, нет. 2. С. 65–77, 2012.

[32] Г. Кляйн и Д. Мюррей, «Повышение гибкости SLAM на основе ключевых кадров», в *Proc. Евро. Конф. Comput. Зрение*, Марсель, Франция, октябрь 2008 г., стр. 802–815.

[33] К. Пиркер, М. Рутер и Х. Бишоф, «CD SLAM-непрерывная локализация и отображение в динамическом мире», в *Proc. IEEE / RSJ Int. Конф. Intell. Роботы Syst.*, Сан-Франциско, Калифорния, США, сентябрь 2011 г., стр. 3990–3997.

[34] С. Сонг, М. Чандракер и К. С. Гест, «Параллельная монокулярная визуальная одометрия в реальном времени», in *Proc. IEEE Int. Конф. Робот. Автомат.*2013. С. 4698–4705.

[35] П.Ф. Алькантарилла, Дж. Нуэво и А. Бартоли, «Быстрая явная диффузия ускоренных элементов в нелинейных масштабных пространствах», представленная в Brit. Max. Vision Conf., Бристоль, Великобритания, 2013.

[36] Х. Yang, К.-Т. Ченг, «LDB: сверхбыстрая функция для масштабируемой дополненной реальности на мобильных устройствах», в *Proc. IEEE Int. Symp. Смешанная дополненная реальность*2012. С. 49–57.

[37] Р. Куэммерле, Г. Гризетти, Х. Страсдат, К. Конолиге и У. Бургард, «g2o: общая структура для оптимизации графов», в *Proc. IEEE Int. Конф. Робот. Автомат.*, Шанхай, Китай, май 2011 г., стр. 3607–3613.

[38] Дж. Штурм, Н. Энгельхард, Ф. Эндрес, У. Бургард и Д. Кремерс, «Тест для оценки систем RGB-D SLAM», в *Proc. IEEE / RSJ Int. Конф. Intell. Роботы Syst.*, Виламура, Португалия, октябрь 2012 г., стр. 573–580.

[39] М. Смит, И. Болдуин, У. Черчилль, Р. Пол и П. Ньюман, «Новое видение колледжа и набор лазерных данных». *Int. J. Робот. Res.*, т. 28, вып. 5. С. 595–599, 2009.

[40] А. Гейгер, П. Ленц, К. Стиллер и Р. Уртасун, «Зрение встречает робототехнику: набор данных KITTI», *Int. J. Робот. Res.*, т. 32, нет. 11. С. 1231–1237, 2013.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.

[41] В. Лепет, Ф. Морено-Ногер и П. Фуа, «EPnP: точное O (n) решение проблемы PnP», *Int. J. Comput. Зрение*, т. 81, нет. 2. С. 155–166, 2009.

[42] BKP Horn, "Решение абсолютной ориентации в замкнутой форме с использованием единичных кватернионов", *J. Opt. Soc. Амер. A*, т. 4, вып. 4. С. 629–642, 1987.

[43] Ф. Эндрес, Дж. Хесс, Дж. Штурм, Д. Кремерс и У. Бургард, «Трехмерное отображение с помощью камеры RGB-D», *IEEE Trans. Робот.*, т. 30, нет. 1. С. 177–187, февраль 2014 г.

[44] Р. А. Ньюкомб, С. Дж. Лавгроув и А. Дж. Дэвисон, «DTAM: плотное отслеживание и картографирование в реальном времени», в *Proc. IEEE Int. Конф. Comput. Зрение*, Барселона, Испания, ноябрь 2011 г., стр. 2320–2327.

[45] С. Лавгроув, Эй Джей Дэвисон и Дж. Ибанес-Гусман, «Точная визуальная одометрия с задней парковочной камеры», in *Proc. IEEE Intell. Транспортные средства Symp.*2011. С. 788–793.

[46] PH Torr и A. Zisserman, «Методы, основанные на признаках для оценки структуры и движения», в *Алгоритмы зрения: теория и практика*. Нью-Йорк, штат Нью-Йорк, США: Springer, 2000, стр. 278–294.

[47] Р. Мур-Артал и Дж. Д. Тардос, «Вероятностное полу-плотное картирование на основе высокоточного монокулярного SLAM на основе характеристик», представленное на Proc. Робот .: Науки. Syst., Рим, Италия, июль 2015 г.

[48] Х. Страсдат, «Локальная точность и глобальная согласованность для эффективного визуального SLAM», доктор философии. диссертация, Имперский колледж Лондона, Лондон, Великобритания, октябрь 2012 г.

Рауль Мур-Арталь родился в Сарагосе, Испания, в 1989 году. Он получил степень промышленного инженера (в области промышленной автоматизации и робототехники) в 2012 году и степень магистра в области систем и компьютерной инженерии в 2013 году в Университете Сарагосы, Сарагоса, где он в настоящее время работает над Кандидат наук. степень в группе робототехники, восприятия и реального времени I3A.

Его исследовательские интересы включают визуальную локализацию и долгосрочное картографирование.

Дж. М. М. Монтьель(M'15) родился в Арнедо, Испания, в 1967 году. Он получил степень магистра и доктора философии. степень в области электротехники Университета Сарагосы, Сарагоса, Испания, в 1992 и 1996 годах, соответственно.

В настоящее время он является профессором Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza, где отвечает за гранты и курсы по исследованиям в области восприятия и компьютерного зрения. Его интересы включают локализацию видения в реальном времени и семантическое отображение для жестких и нежестких сред, а также распространение этой технологии.

в области роботизированных и нероботических приложений.

Доктор Монтиэль является членом группы I3A по робототехнике, восприятию и работе в реальном времени Университета Сарагосы. Он был награжден несколькими испанскими грантами MEC для финансирования исследований с Оксфордским университетом, Великобритания, и с Имперским колледжем Лондона, Великобритания.

Хуан Д. Тардос (M'05) родился в Уэске, Испания, в 1961 году. Он получил степень магистра и доктора философии. дипломы по электротехнике Университета Сарагосы, Сарагоса, Испания, в 1985 и 1991 годах соответственно.

Он является профессором Departamento de Informática e Ingeniería de Sistemas Университета Сарагосы, где отвечает за курсы робототехники, компьютерного зрения и искусственного интеллекта. Его исследовательские интересы включают одновременную локализацию и картографирование, восприятие и мобильную робототехнику.

Доктор Тардос является членом I3A Robotics, Perception и Real-Time Group, Сарагосский университет.

Разрешенное лицензионное использование ограничено: IEEE Xplore. Загружено 23 декабря 2021 года в 10:26:14 UTC с сайта IEEE Xplore. Ограничения применяются.