

Подпишитесь на DeepL Pro для редактирования данного документа.  
Дополнительную информацию можно найти на странице [www.DeepL.com/pro](https://www.deepl.com/pro?cta=edit-document).

**2018 IEEE/RSJ Международная конференция по интеллектуальным роботам и системам (IROS) Мадрид, Испания, 1-5 октября 2018 г.**

Локализация стереокамеры в 3D картах LiDAR

Ёнджи Ким1, Джинионг Чонг 1и Аён Ким 1∗

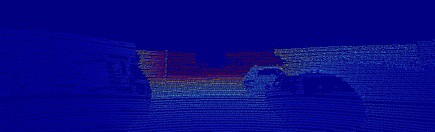
***Аннотация:* С появлением 3D-датчиков Light Detection and Ranging (LiDAR) технологии одновременной локализации и картографирования (SLAM) получили широкое распространение, и точные 3D-карты стали легко доступны. Многие исследователи обращают свое внимание на локализацию на ранее полученной 3D-карте. В этой статье мы предлагаем новый и легкий алгоритм визуального позиционирования только с помощью камеры, который включает в себя локализацию в предыдущих 3D картах LiDAR. Мы стремимся достичь точности системы глобального позиционирования (GPS) потребительского уровня, используя зрение в городской среде, где сигнал GPS ненадежен. Используя стереокамеру, глубина из карты диспаратности стереосистемы сопоставляется с 3D картами LiDAR. Полная позиция камеры с шестью степенями свободы (DOF) оценивается путем минимизации остаточной глубины. Благодаря визуальному отслеживанию, которое обеспечивает хорошее начальное предположение для локализации, предлагаемый остаток глубины успешно применяется для оценки положения камеры. Наш метод работает в режиме онлайн, а средняя ошибка локализации сопоставима с ошибками, полученными с помощью современных подходов. Мы проверили предложенный метод в качестве автономного локализатора с помощью KITTI.**

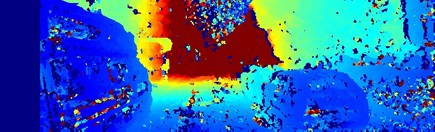
Камера

LIDAR

Остаточный

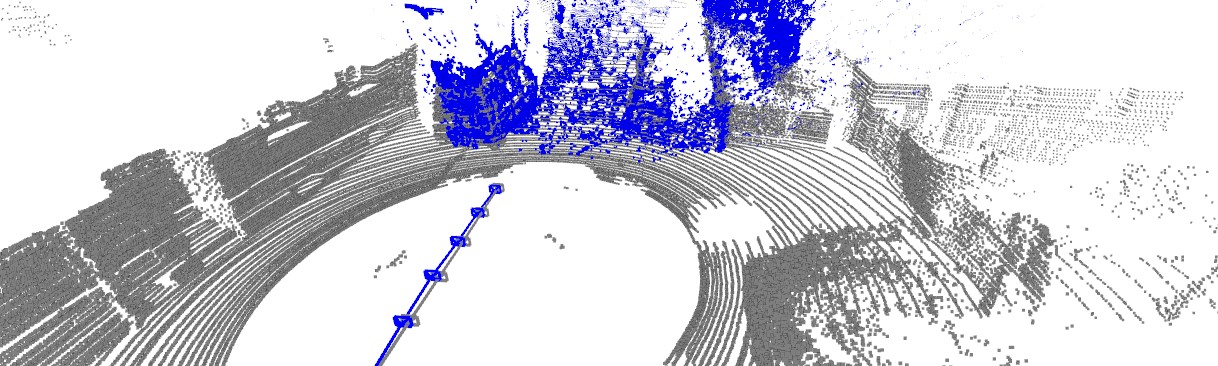
Интенсивность

Высокий



Глубина

Низкий



**и как модуль в системе SLAM с использованием нашего собственного набора данных.**

1. ВВЕДЕНИЕ

Локализация очень важна для роботизированных систем, таких как самоуправляемые автомобили. Все остальные навигационные задачи (например, планирование и управление) выполняются на основе точной оценки положения робота. Традиционно автономные транспортные средства полагаются на GPS/инерциальную навигационную систему (ИНС) для локализации [1], [2], [3]. GPS обеспечивает глобальное положение без дрейфа, которое может быть объединено с высокочастотной относительной позой, оцененной INS. Однако GPS часто страдает от периодических ошибок, вызванных многолучевыми эффектами. Типичными примерами являются городские каньоны и внутренние помещения, где точная оценка позы ограничена.

Несмотря на наличие точной карты [4], [5], [6], локализация по этим картам требует регистрации между картой и бортовыми датчиками. Очевидным и предпочтительным методом было бы использование одного и того же датчика для картографирования и локализации. Это направление исследований было найдено в [7], [8], [9], в которых исследователи использовали LiDAR для 3D-картирования и локализации. Поскольку LiDAR обеспечивают точные данные о дальности в 3D, прямое сопоставление между заданной картой и текущим сканом было возможно с использованием методов сопоставления сканов, таких как Iterated Closest Point (ICP) [10]. Однако, из-за связанных с этим затрат и физических требований, исследователи предпочитают добиваться локализации с помощью зрения [11], [12], [13], [14],

[15].

Основная дилемма локализации камеры на картах LiDAR возникает из-за различий в модальностях камеры и

Y. Ким, Дж. Чон и А. Ким работают на кафедре гражданского и экологического строительства, KAIST, Тэджон, С. Корея [youngjikim, jjy0923, ayoungk]@kaist.ac.kr

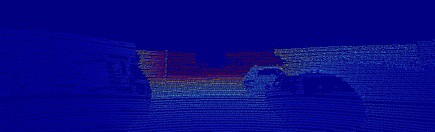
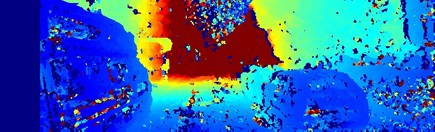
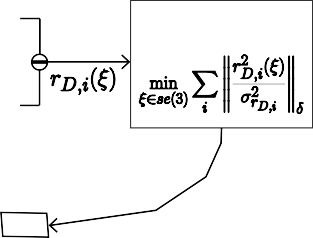
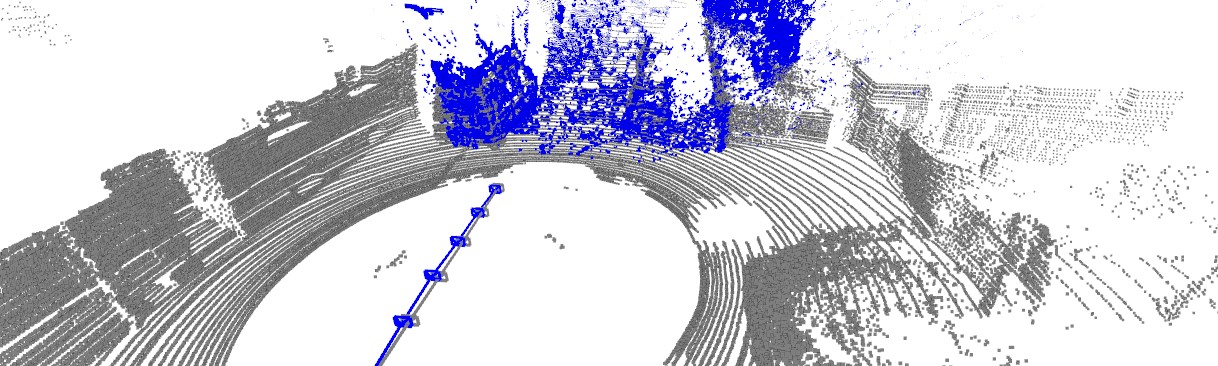
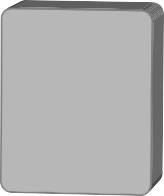
Рис. 1. Обзор предлагаемого алгоритма. Вместо того, чтобы сравнивать камеру

и интенсивности LiDAR, мы используем стереоглубину и сравниваем ее с глубиной на карте LiDAR. Поза камеры оценивается путем минимизации остаточной глубины. На рисунке ниже показана расчетная позиция на карте, где серые точки - это точки карты, а синие - точки, реконструированные с помощью стереокамеры. Траектория камеры, обозначенная синими линиями, повторяет траекторию наземной камеры, обозначенную серыми линиями.

Данные LiDAR. На рис. 1 показано, как различается информация, предоставляемая этими двумя датчиками. Данные LiDAR состоят из диапазонов и значений отражательной способности, в то время как камеры предлагают изображения интенсивности в цвете или сером масштабе. Судя по остаткам на рис. 1, значения отражательной способности, полученные с помощью LiDAR, в основном не совпадают со значениями интенсивности, полученными с помощью камеры. Глубины, оцененные с помощью стереокамер и LiDAR, более согласованы, но все еще имеют расхождения на краях. Кроме того, плотность данных, полученных с помощью двух датчиков, различна. Типичные изображения LiDAR показывают плотность полос, которые по горизонтали плотные, а по вертикали разреженные, в то время как камеры дают изображения с равномерной плотностью.

Последние подходы к локализации камер на 3D картах были направлены на преодоление проблем, вызванных различиями, присущими данным камер и LiDAR. Одним из наиболее распространенных подходов является фотометрия. Уолкотт и Юстис [12] использовали изображения отражения LiDAR и сопоставили их с изображением интенсивности камеры. Предварительно было подготовлено несколько изображений отражения, которые могут быть сопоставлены с изображением камеры. Поза камеры соответствует синтезированному изображению, которое имеет максимальную нормализованную взаимную информацию (NMI), выбранную в качестве текущей позы камеры. Другой подход, предложенный Стюартом и Ньюманом [11], использовал нормализованное информационное расстояние (NID) в качестве метрики для согласования интенсивностей камеры и LiDAR. Вместо синтеза изображений-кандидатов LiDAR и наивного поиска методом перебора, они решили проблему с помощью следующих методов

**978-1-5386-8094-0/18/$31.00 ©2018 IEEE5826**



Предварительная обработка

карта

Исходная поза

стереопоток

слева

правильно

Карта глубин

Поколение

Отслеживание

Оценить преобразование между последовательными кадрами изображения с помощью

Местная карта

Извлечение

Глубина LIDAR

Локализация

Оценка положения камеры на карте путем минимизации остатков глубины

глубинаОценка камеры

стерео глубинарезидуалы позируют через

Рис. 2. Схема предлагаемой системы. Входными данными алгоритма являются предварительная карта 3D LiDAR, начальная позиция камеры и поток стереоизображений. Предварительная обработка включает извлечение локальной карты и создание карты глубины. При отслеживании позиция камеры между последовательными кадрами левого изображения оценивается и используется в качестве начального предположения для локализации. Для локализации мы находим положение камеры, при котором остаточная глубина минимизируется.

Квази-ньютоновская оптимизация с аналитическими производными. Это позволяет напрямую оценивать положение камеры 6 DOF.

Вторая стратегия локализации камеры на 3D-картах заключается в использовании геометрии. После восстановления точек, наблюдаемых с камеры, с помощью визуальных схем SLAM, таких как корректировка пучков, проблема теперь заключается в сопоставлении двух наборов облаков точек. Форстер *и др*. [16] предложили метод выравнивания 3D-карт, полученных с двух разных платформ, оснащенных датчиком глубины и монокулярной камерой. Поза камеры в пределах карты оценивалась путем сканирования, совмещая 3D-карту, полученную датчиком глубины, и точки, реконструированные в результате корректировки пучка. Аналогично, в [13] облака точек камеры, полученные с помощью ORB- SLAM, использовались для сопоставления с априорной картой. Для сопоставления облаков точек с различной плотностью был предложен метод фильтрации и связывания соответствующих точек с помощью вокселей. Гибрид фотометрического и геометрического сопоставления является одним из перспективных методов для локализации камеры в пределах заданной 3D карты. Нойберт *и др*. [14] извлекли изображения глубины из предварительной карты и сопоставили их с изображениями интенсивности камеры. И наоборот, Сюй *и др*. [15] извлекли изображения глубины из стереокамеры. Они синтезировали изображения отражения LiDAR на основе карты и сравнили их с изображениями глубины. Мы предлагаем новую гибридную схему сопоставления для локализации стереокамеры на заданной предварительной карте. В отличие от предыдущих гибридных методов, которые пытаются сопоставить геометрическую и фотометрическую информацию, мы в первую очередь используем геометрическую информацию, поскольку она более последовательна, чем фотометрическая, как показано на рис. 1. Однако фотометрическое сопоставление с использованием геометрии осуществляется путем выравнивания изображений, полученных с помощью информации о глубине. После получения изображений глубины со стереокамеры и заданной карты можно оценить положение камеры, минимизировав разницу в двух изображениях глубины. Оценка позы путем уменьшения остатка глубины также описана в методах прямого монокулярного SLAM, таких как LSD- SLAM [17], который использует остаток глубины для компенсации дрейфа.

при выравнивании ключевых кадров. Вместо использования остатка глубины для выравнивания ключевых кадров мы стремимся совместить текущее измерение камеры и заданную карту. Минимизация остатка глубины приводит к успешной локализации стереокамеры.

Предлагаемый подход к локализации представляет собой существенное улучшение по сравнению с предыдущими методами. Вместо того чтобы находить по одной DOF за раз или предоставлять только 3 DOF позу, как в [12], [14], [15], наш метод ищет полную 6 DOF позу камеры одновременно. Более того, наш алгоритм требует меньше вычислительных затрат, поскольку нет необходимости синтезировать изображения из карты. Действительно, синтез изображений требует больших вычислений. В предыдущих исследованиях для синтеза изображений использовался GPU [12], либо синтетические изображения готовились заранее в автономном режиме [14]. Другим узким местом в алгоритме является вычисление метрики для сопоставления. NMI, используемый в [12], или NID в [11] имеют сложность алгоритма *O*(*n* + *m*2 ), где *n* - количество наблюдаемых 3D точек, а *m* - количество бинов интенсивности. Наша метрика представляет собой сумму разностей глубин со сложностью *O*(*n*).

Наш метод похож на [13] в том, что полная камера с 6 DOF

позиция вычисляется путем сопоставления геометрии. Однако, поскольку мы используем стереокамеру, а не монокулярную камеру, нет необходимости оценивать масштаб сцены в дополнение к 6 DOF позе камеры. Сократив пространство поиска с 7D до 6D, мы можем получить более высокую точность локализации. Кроме того, вместо проведения ICP, наш метод обеспечивает фотометрическое соответствие путем минимизации остатка глубины. Основное преимущество использования остатка глубины заключается в том, что исключается фаза поиска соответствия, которая многократно выполняется в ICP. Поиск соответствия является громоздкой задачей, особенно когда плотности двух совпадающих облаков точек не совпадают. В [13] эта проблема была решена путем фильтрации облака точек с использованием вокселей. Наш метод минимизирует избыточные процессы, что позволяет применять его к 3D картам с различной плотностью.

Таким образом, мы предлагаем локализацию стереокамеры.

**5827**

метод в 3D LiDAR картах, который имеет следующие атрибуты:

* Мы решаем задачу мультимодального сопоставления путем минимизации остаточной глубины.
* Наш алгоритм не ориентирован на глобальную локализацию. Нам необходимо задать начальное положение камеры на карте и оценить относительное положение камеры между сценами с помощью визуального отслеживания.
* Одновременно мы оцениваем полную позу камеры с 6 DOF.
* Предложенный метод является легким и работает в режиме онлайн на центральном процессоре.

1. ПРЕДЛОЖЕННЫЙ МЕТОД

Мы предлагаем систему, способную локализовать стереокамеру относительно предварительно заданной 3D-карты. Мы предполагаем, что исходная позиция камеры задана, и выполняем локализацию с учетом грубого начального предположения. На рис. 2 показана схема предлагаемого локализатора. Наша система состоит из четырех модулей. В процессе предварительной обработки исходные данные, полученные с карты и потока стереоизображений, обрабатываются для использования в следующем модуле отслеживания и локализации. При генерации карты глубины, карта глубины генерируется с помощью стереодиспаратности. При извлечении локальной карты из глобальной карты извлекается локальная 3D-карта, которая будет сопоставлена с картой глубины. Для определения начального предположения о положении камеры перед локализацией добавляется отслеживание. В этом модуле оценивается относительная поза между последовательными кадрами изображения. Затем позиция камеры 6 DOF оценивается с помощью локальной карты, карты глубины и предположительной позиции, полученной в процессе отслеживания.

В рамках предлагаемого метода мы используем однородные координаты точек в 2D и 3D, приведенные ниже.

карта. Выделение локальной карты осуществляется с помощью восьмеричного дерева [18]. Октри - это древовидная структура данных, обычно используемая для хранения трехмерных облаков точек. Каждый узел в octree имеет восемь дочерних элементов, которые представляют собой восемь подкубов. Быстрая разбивка пространства и поиск осуществляются с помощью восьмеричного дерева. Библиотека облаков точек (PCL) [19] предоставляет восьмерицу и различные методы поиска соседей. Для извлечения точек, окруженных текущим положением камеры, используются соседи в пределах радиуса поиска для извлечения локальной карты.

*B. Генерация карты глубин*

Основная стратегия заключается в использовании карты глубины из текущего стереопотока. Построенная карта глубины в основном используется в предстоящем отслеживании и локализации. На первом этапе создания карты глубины, карта диспаратности создается с помощью стерео полуглобального сопоставления блобов (SGBM), предоставляемого OpenCV [20]. SGBM [21] - это метод стереосопоставления, который оценивает диспаратность путем минимизации энергетической функции, состоящей из взаимной информации, основанной на стоимости пикселя и стоимости глобальной гладкости. С помощью SGBM можно получить плотную стереокарту диспаратности. Затем глубина сцены оценивается с помощью метода [22], который устраняет статистическую погрешность триангуляции в зависимости от расстояния.

*C. Визуальное отслеживание*

Визуальное слежение обеспечивает начальное предположение для локализации. Ниже приведен простой алгоритм визуального слежения, основанный на минимизации фотометрической ошибки по методу Гаусса-Ньютона. Когда локализатор используется как отдельный модуль, мы применяем этот простой алгоритм отслеживания. Однако мы оставляем выбор алгоритма визуального отслеживания на усмотрение пользователя.

*x*

*x*

*y*

поскольку локализатор, основанный на глубине, работает независимо от выбора отслеживания. Например, для проверки в пределах

**x** =y

1

и **p** = *z*

1

(1)

SLAM, мы приняли модуль отслеживания ORB-SLAM.

[23] для тесного взаимодействия с системой ORB-SLAM. Относительная поза между предыдущим кадром и

*n*

Поза камеры представлена с помощью преобразования *SE*(3) в виде

текущий кадр **T***n*−1 оценивается путем минимизации энергии

**T** = **Rt**

01

*,* где **T ∈** *SE*(3)*,* (2)

функция, суммирующая все пиксели *i,* заданная как

Σ ¨ *r2*

*σ*

*(*ξ) ¨

2

которая связана с соответствующей *се*(3) алгеброй Ли *ξ* ∧

по экспоненциальной карте

*E(*ξ) =

*I,i*

*i* ¨rI *,i*

¨

*.* (5)

*δ*

**T***(*ξ) = exp*(ξ* ∧)*.* (3)

Оператор клина превращает *ξ* R6 в член алгебры лжи *se*(3). Для отслеживания и локализации позиция камеры **T** итеративно обновляется как

∧∈

**T ← T***(*ξ) - **T***,* (4)

где приращение **T***(*ξ) применяется с использованием левого мультипликатора.

*A. Извлечение локальной карты*

В глобальных картах, содержащих крупномасштабные облака точек, нам необходимо выделить локальную область точек глобальной карты, которую можно наблюдать с текущего положения камеры. Мы определяем эту часть точек глобальной карты как локальную область

Функция энергии определяется как сумма ошибки

residuals over intensity *rI* , divided by their variances, *σ*2 . The Huber norm *δ* lets the energy function converge well via ignoring outliers. We use the difference in image intensities as the error residual, which is given as

*r I*

ǁ - ǁ

*rI,i* *(*ξ) = *In*−1 π T*(ξ)***T** *n*−1- π −1(**x** *i, d* *i*) - *I n*[**x** *i*]*.* (6)

*n*

В этом уравнении **x***i* = [*ui* *, vi* *,* 1]T - трехмерный вектор, представляющий собой координаты изображения *i-го* thпикселя.

в однородных координатах. Глубина соответствует

*i-й*th пиксель обозначается как *di* . Соответствующая точка в трехмерном пространстве, **p** = [*x, y, z,* 1]T, проецируется на плоскость изображения с помощью функции проекции изображения π( - ) как

**x** = π(**p**)*.* (7)

**5828**



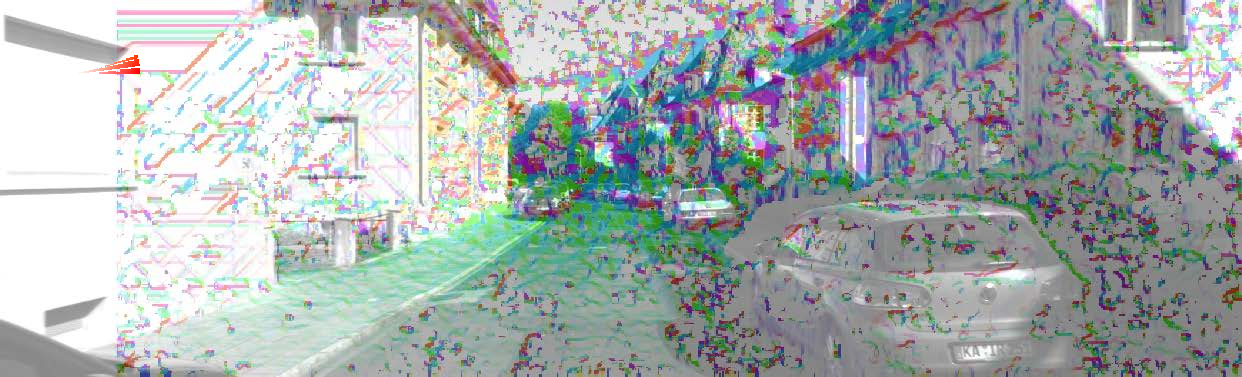
u

v

z

x

y



(a) Три плоскости в образце сцены(b) Наложенный градиент глубины

Рис. 3. Образец сцены и ее градиенты глубины. (a) показывает образец сцены и координаты камеры. Три ортогональные плоские поверхности (зеленая, желтая и белая) и их ожидаемые градиенты глубины показаны стрелками. Ожидается, что градиент глубины будет преобладать в направлении u для желтой плоскости и в направлении v для зеленой плоскости. Белая плоскость имеет небольшой градиент глубины, и сильного значения не ожидается. (b) иллюстрирует вычисленный градиент глубины, наложенный на образец сцены. Цветовая карта в левом верхнем углу представляет собой двумерный векторный цветовой код. Градиент глубины, связанный с каждым пикселем, следует за цветом в зависимости от его направления. Вычисленные градиенты глубины, относящиеся к каждой плоскости, согласуются друг с другом, отображая один и тот же цвет. Более того, для пикселей, принадлежащих трем ортогональным плоскостям, ожидаемые градиенты глубины правильно согласуются с вычисленным градиентом глубины в (b).

И наоборот, π−1 ( - ) обозначает инверсию проекции изображения.

остаток глубины составляет

функции. Интенсивности изображений предыдущего и текущего

**J** = [0*,* 0*,* 1*,* 0](**T***C* **p** )Ⓢ *- ∂D*(**xi**) - ∂π(**pi**)(**T** *C***p** )*Ⓢ,*

кадры задаются как *In*−1 и *In* . Таким образом, I[**x**] обозначает *i*

интенсивность точки изображения в точке **x**.

*M i ∂x i*

*∂p i*

*M i*

(11)

Мы решили задачу нелинейной оптимизации, используя *g2owhere* производную проективной функции камеры [24], общую структуру оптимизации графов. Начиная с∂π (**p** *i*)*/∂p* *i*задается как

хорошее начальное предположение, относительное преобразование, **T***n*−1, является

*f*  *f*

*z*

-

обновлено через

*n x*

*∂π*(**p** *i*)

**T***n*−1 ← **T***(*ξ) - **T** *n*−1*,* (8)

=

0f *y*

0 *xx* 0

*z2*

*f y*

-y0

(12)

*n n ∂p*

*z z2*

где приращение *ξ* вычисляется из

**J ΩJξ**T= -J **Ω** TT*r*(0)*.* (9)

Здесь **J** состоит из **J** , матрицы Якобиана от

0001

так как мы предполагаем проекционную модель камеры. Градиент глубины ∂D(**x***i* )*/∂xi* может быть получен путем применения фильтрации Шарра к карте глубины.

*с*

*i*

остаток *rI,i* . Информационная матрица Ω состоит из

Разнообразие в этом градиенте глубины важно для ло-

производительность калибровки. Эта взаимосвязь может быть понята

диагональные члены с обратными дисперсиями ошибок.

*D. Локализация*

Визуальная локализация выполняется путем сопоставления точек карты и текущей карты глубины, полученной от стереокамеры. Хотя функция объективной энергии идентична (5), остаточная глубина *r* *D*является такой, как описано ниже.

*rD,i* *(*ξ) = [**T***(*ξ)**T** *C*- **p** *i*](3) - D(π(**T***(*ξ)**T** *C*- **p***i* )) (10)

на примере сцены с небольшим остатком глубины. На рис. 3 показан образец сцены и градиент глубины, который мы вычислили из этой сцены с помощью фильтрации Шарра. Каждый пиксель на рис. 3(b) окрашен в зависимости от направления градиента глубины. Из рис. 3 мы можем понять, как геометрия, характеризуемая градиентом глубины, влияет на общий процесс оценки положения камеры. Например, когда градиент по оси u больше, чем по оси v, как в случае с желтым прямоугольником, это наблюдение

*ММ*

в основном влияет на обновление позы в направлении x. Это

Этот остаток по глубине оптимизируется в модуле локализации вместо ранее принятого остатка по интенсивности

(6) на этапе отслеживания. Остаток глубины определяется как разница между глубиной точки карты и соответствующей стерео глубиной. После преобразования точки карты **p** *i*в координаты камеры через **T** *C*, глубина точки карты может быть получена как [**T***(*ξ)**T** *C***p** *i*](3), обозначая значение оси z преобразованной точки карты. Для получения соответствующей глубины стереоизображения точка в координатах камеры проецируется на координаты изображения с помощью функции проекции камеры π( ), а затем глубина получается с помощью отображения глубины ( ).

*M*

*M*

-

D -

-

Затем оптимизация выполняется аналогично тому, как она выполняется в модуле слежения. Матрица Якобиана

потому что якобиан, в котором доминирует градиент оси u, имеет большой первый элемент, что способствует обновлению по оси x. Аналогично, когда доминирует градиент оси v, как в случае с зеленым прямоугольником, обновляется поза в направлении y. Если градиент оси u или оси v мал, как в случае третьего прямоугольника, обновляется позиция камеры по оси z.

В свете этого примера мы видим, что наблюдения с различными градиентами глубины подходят для нашего метода локализации. Однако слишком сложные среды, полные резких различий в глубине, создаваемых трехмерными гранями, препятствуют сбалансированной оценке позы. Это происходит потому, что точное положение граней, оцененное по стереоглубине, иногда может быть неверным, но градиент глубины, вызванный гранями, облегчает работу.

**5829**

500

450

400

1. Ошибки локализации (среднее ± среднее)

1

0.8

трансляционная ошибка (м)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Последовательность # | Translational | | ошибка [m] | Погрешность вращения [◦] | |
| 00 | 0.1325  0.2205  0.2368  0.4496  0.1462  0.3753  0.1305  0.1440  0.1799  0.2398 | ±  ±  ±  ±  ±  ±  ±  ±  ±  ± | 0.1100  0.2918  0.2267  0.3537  0.1427  0.6653  0.1157  0.1755  0.2165  0.6767 | 0.3221  0.3262  0.4133  0.8758  0.3402  0.8485  0.4872  0.3279  0.3375  0.4934 | ± 0.3911  ± 0.5355  ± 0.3286  ± 0.3857  ± 0.4033  ± 1.8009  ± 0.4718  ± 0.5453  ± 0.3440  ± 0.7690 |
| 02 |
| 03 |
| 04 |
| 05 |
| 06 |
| 07 |
| 08 |
| 09 |
| 10 |

0.6

0.4

0.2

0

1

0.8

трансляционная ошибка (м)

0.6

0.4

0.13246 0.10996

0102030405060708090100

350

0.2

300

250

y (м)

200

150

0

010

5

0.32206 0.3911

4

погрешность вращения ( °)

2030

латеральный

вертикальный

продольный

405060

7080

90100

3

GT

Реквизиты

100

502

01

-50

-300-200-1000100200300

x (м)

1. Траектории

0

0102030405060708090100

1. Ошибки локализации

Рис. 4. Результат на KITTI 00. (a) суммирует ошибку локализации в среднем со стандартным отклонением. (b) показана траектория "наземной истины" и расчетная траектория, полученная с помощью нашего алгоритма локализации. Три графика в (c) демонстрируют ошибки локализации в процентном отношении к траектории. Первый график показывает трансляционную ошибку, которую можно разделить на трансляционные ошибки в боковом, вертикальном и продольном направлениях, как на втором графике. Вращательная ошибка показана на третьем графике. Пунктирная линия на первом и третьем графике указывает на среднее значение ошибки, так как среднее стандартное отклонение отображается в верхней части графика.

±

внезапное изменение позы. Таким образом, наша стратегия заключается в том, чтобы сосредоточиться на плоскостях, положение которых мы можем надежно оценить. Для этого мы задаем вариации остаточной глубины пропорционально величине градиентов глубины, чтобы края меньше влияли на общую энергию.

1. ЭКСПЕРИМЕНТ

Мы проверили предложенный метод на двух различных наборах данных. Мы использовали набор данных одометрии KITTI [25] и сравнили эффективность нашей локализации с результатами недавней работы Caselitz *et al.* [13]. Чтобы доказать осуществимость предложенного алгоритма в более реалистичной среде, мы дополнительно применили алгоритм на наборе данных, собранных нашей платформой городского картографирования. [26]. С результатами локализации на обоих наборах данных можно также ознакомиться в прилагаемом мультимедийном файле camloc.mp4.

Мы проверили предложенный метод в двух сценариях, как

независимый модуль локализации и как модуль в системе SLAM. Сначала мы проверили наш модуль визуальной локализации путем тестирования на общедоступном наборе данных KITTI. Каждая визуально локализованная поза сравнивается с базовой линией, представленной в KITTI, для количественной оценки. Для второй оценки мы встроили наш локализатор в систему ORB SLAM. Для бесшовной интеграции мы импортировали модуль отслеживания из ORB SLAM в качестве инициализатора для локализатора.

1. *Оценка на наборе данных KITTI*

Набор данных одометрии KITTI содержит последовательности изображений, полученных с помощью синхронизированной стереокамеры, сканов Velodyne HDL- 64E LiDAR и поз наземной истины. В наборе данных приведена внешняя калибровка между камерой и LiDAR, а также внутренняя/внешняя калибровка для стереокамеры. Мы оценили наш алгоритм на предоставленных последовательностях с истинными позами (последовательности 00-10), за исключением последовательности 01. Поскольку последовательность 01 - это последовательность шоссе, которая содержит ограниченное количество особенностей сцены, наш алгоритм отслеживания не сработал в этой последовательности. Поскольку мы больше сосредоточены на городских районах, где GPS ненадежен, мы рассматриваем другие последовательности в оценке. Мы выполнили локализацию, используя предварительную карту, начальное положение камеры на карте и последовательность стереоизображений. Даже если глобальная карта должна быть подготовлена до локализации, мы можем обойти процесс составления карты, используя исходные позы роботов. Сканы LiDAR, полученные из последовательных поз робота, были преобразованы в глобальные координаты через соответствующие позы наземного робота и использованы в качестве локальных карт. Мы оценивали точность локализации, вычисляя разницу между оцененной позой камеры и

позиция истины на местности.

На рис. 4 показан результат локализации с помощью KITTI

**5830**

последовательность 00. Траектория камеры, оцененная нашим алгоритмом ло-кализации, в основном похожа на траекторию наземной истины. На графиках ниже показаны трансляционные и ротационные ошибки вдоль последовательности. Ошибка трансляции составляет

всегда меньше 1*,*0 м, а ошибка вращения меньше 5◦. Средняя ошибка трансляции составляет 0*,*13 м, а средняя ошибка вращения - 0,3◦. Когда мы проанализировали ошибку

графиков, мы обнаружили, что более значительная ошибка произошла на перекрестке

где камера вращается по мере поворота автомобиля, а соседние здания и припаркованные автомобили исчезают из сцены. Таким образом, мы делаем вывод, что предложенный алгоритм не применим к ситуации с быстрыми вращательными движениями и без близлежащих структур. Однако производительность предложенного метода сравнима с другими методами. Наш результат для последовательности 00 показывает более низкие средние ошибки трансляции и вращения, чем результат, предложенный в [13]1.

В таблице на рис. 4(a) приведены ошибки локализации для других последовательностей KITTI. В этой таблице ошибки усреднены по всем позам в каждой последовательности и представлены в виде среднего значения

±

формат стандартного отклонения. Средние ошибки трансляции составляют менее 0,5 м, а средние ошибки вращения - менее 1,0◦. Наш алгоритм показывает наилучшую производительность на последовательности 00, в то время как

наихудшие показатели у последовательности 04. Мы предположили, что ограниченной количество структур в сцене может быть основной причиной снижения производительности.

1. *Оценка на нашем наборе данных*

Оценка набора данных KITTI выявила субметровую точность калибровки. Однако в наборе данных KITTI большинство сцен было снято в жилых районах или на автомагистралях, с одной полосой, окруженной малоэтажными зданиями. Кроме того, временная разница между этапами картирования и локализации была недостаточно хорошо учтена, так как мы локализовали по одной и той же последовательности. Для дальнейшей оценки нашего алгоритма в более сложных реальных сценариях и проверки метода в качестве модуля локализации в системе SLAM, мы провели второй эксперимент с использованием нашего собственного набора данных, интегрировав локализатор в ORB-SLAM. Целевая среда является сложной из-за значительной структурной вариации, широких дорог, изменений окружающей среды во времени и динамических объектов. На рис. 8(b) показано количество

1В [13] средние трансляционные и вращательные ошибки последовательности 00 KITTI составили 0,30 м и 1,65◦.



**LiDAR (3D) RTK GPS LiDAR (3D)**

**LiDAR (2D)**

**Камера**

**0.48m**

**Камера**

* 1. Мобильная картографическая система

Рис. 5. Стереокамера, установленная на нашей картографической платформе.

спутников, существенно различающихся в зависимости от местоположения, что указывает на структурные различия.

1. *Экспериментальная установка:* Наша сенсорная система, описанная в

[26] был оснащен 3D и 2D LiDAR вместе с навигационными датчиками. Для создания предварительной карты мы использовали базовую траекторию, представленную в наборе данных. 3D LiDAR карта была восстановлена на основе этой базовой траектории. Для локализации мы дополнительно установили стереокамеру на передней стороне автомобиля, как показано на рис. 5. Для стереокамеры мы установили две камеры PointGrey Flea3, которые обеспечивают изображение с разрешением 1384 x 1032 при 18 кадрах в секунду. Две камеры были установлены лицом вперед, как показано на рис. 5, с расстоянием между ними 0,48 метра.

1. *Внутренняя/внешняя калибровка:* Мы оценили внутренние и внешние параметры стереокамеры, используя приложение MATLAB для калибровки стереокамеры. Кроме того, чтобы определить относительное положение стереокамеры в системе датчиков, мы провели внешнюю калибровку между левой камерой и 3D LiDARs. Поскольку стереокамера наблюдает спереди, что противоположно точке зрения LiDARов, прямое соответствие между камерой и LiDARами было невозможным. Вместо этого мы использовали метод из [27], который сравнивал изображение интенсивности камеры с изображением интенсивности LiDAR, синтезированным из частичной карты, реконструированной по движению автомобиля. Относительная позиция между камерой и LiDAR была получена путем минимизации NID, который измерял несходство между двумя изображениями. На рис. 6 показан результат внесистемной калибровки.
2. *Получение данных:* Чтобы применить наш алгоритм в более реалистичном сценарии, мы получили два набора данных для одной и той же местности в разные даты (2018-04-11 и 2018-07-13). Мы создали карту по одному набору данных и протестировали наш алгоритм локализации на карте, созданной по другому набору данных. 3D-карта, построенная на основе набора данных 2018-07-13, представлена на рис. 7. Для проверки SLAM с использованием предложенной визуальной локализации использовался набор данных 2018-04-11. В этом наборе данных автомобиль проехал 15*,*7 км, сделав 34 075 стереоизображений в пределах карты. Мы снова использовали SLAM на основе графа поз для генерации истинных поз камеры, которые служили в качестве базового уровня для оценки производительности системы.

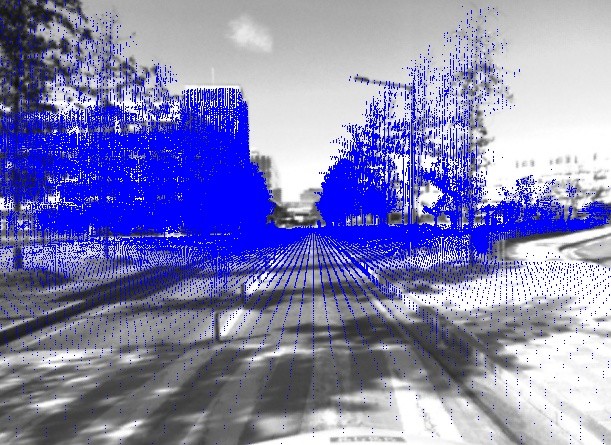


Рис. 6. Результат калибровки камеры и LiDAR. Изображение интенсивности камеры накладывается на точки LiDAR с помощью оценки относительного положения между двумя датчиками.

**5831**

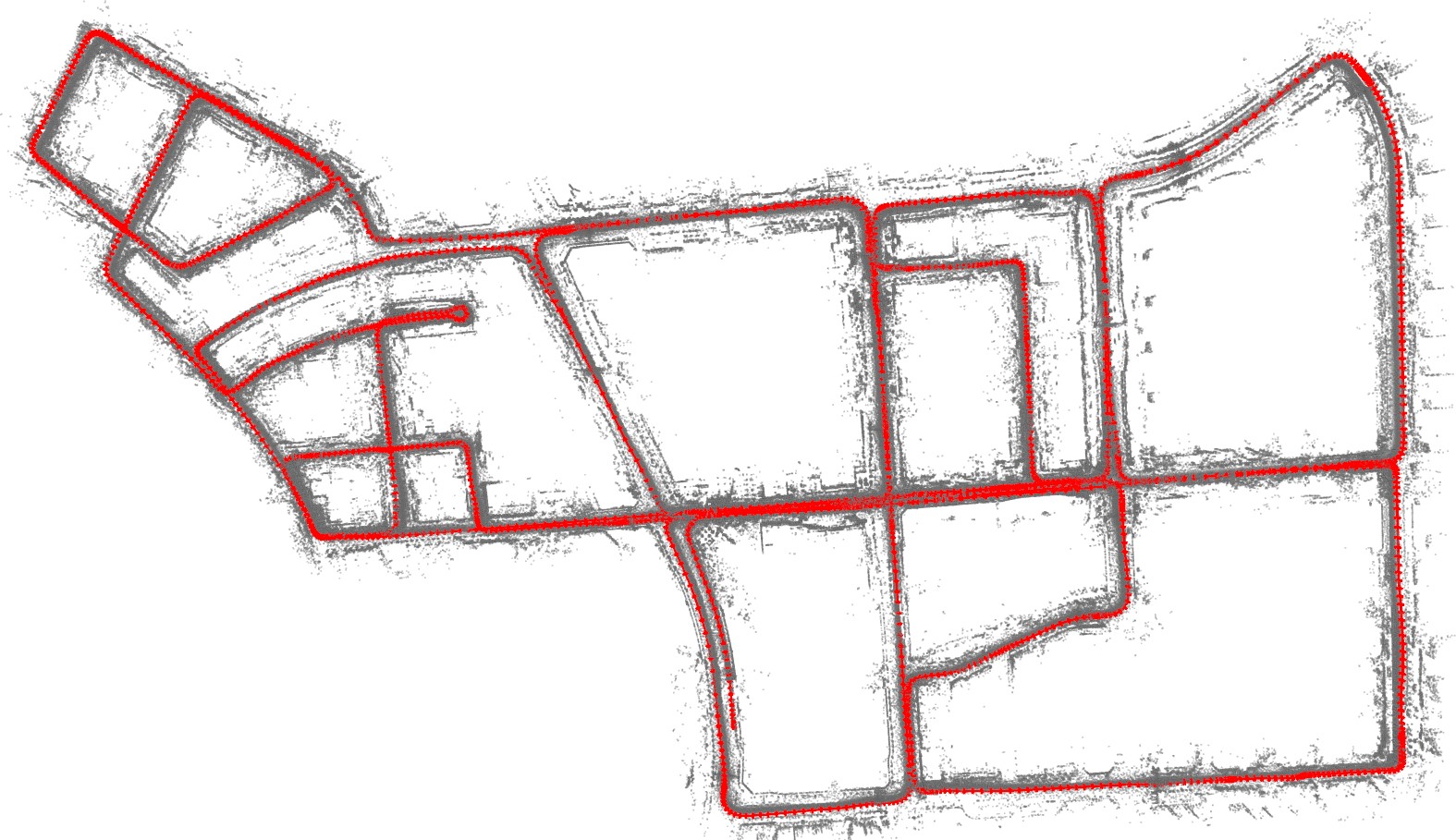


Рис. 7. Глобальная карта (серый) и истинные положения камер (красный), полученные с помощью нашей собственной сенсорной системы.

ТАБЛИЦА I

СТАТИСТИКА ПО КАЖДОМУ МОДУЛЮ

|  |  |  |
| --- | --- | --- |
| Модуль | Локализатор | Местный МЦП |
| Время вычислений [мс] | 27 | 1,813 |
| # кадров | 2,404 | 1,459 |

локализации. Красная траектория на рис. 7 изображает траекторию движения камеры в пределах предварительной карты.

1. *SLAM с использованием локализатора по глубине:* Во время SLAM мы в основном запускали локализатор по глубине в режиме онлайн, выборочно выполняя локальный ICP между восстановленным облаком точек из стереосистемы и предварительной картой только тогда, когда остаток глубины был большим. Локальный ICP требует больших вычислительных затрат, как видно из табл. I, но наше выборочное использование этого модуля позволило обеспечить онлайн производительность всей системы.
2. *Оценка:* Мы сравнили оцененную траекторию движения камеры с траекторией, полученной вместе с предварительной картой. Наложение двух траекторий, как на рис. 8(a), дает качественную оценку производительности. Обратите внимание, что расчетная траектория была получена только с помощью зрения, а согласованность между двумя траекториями сохраняется (т.е. траектория, оцененная с помощью визуальной локализации и наземной истины). Графики ошибок на рис. 8(c) дают трансляционные и вращательные ошибки на траектории. Среднее значение

ошибка при переводе составляет 5*,*34 м, а средняя ошибка при вращении - 4,95◦.

Несмотря на то, что локализатор в SLAM работал надежно

в большинстве регионов, сбои локализации происходили в бесструктурных регионах. Наиболее сложным для локализатора на основе глубины был случай, когда дороги были широкими, а информация о близлежащих структурах была скудной. Мы перезапускали систему каждый раз, когда обнаруживался сбой локализации, аналогично тому, как это было сделано в [28]. Всего было использовано 28 перезапусков из 15*,*7 км длины пути. Пурпурные линии на рис. 8(a) и рис. 8(c) показывают точки перезапуска.

Несмотря на то, что результаты нашего набора данных показали более низкую точность, чем результаты набора данных KITTI, этот эксперимент продемонстрировал осуществимость нашего алгоритма в реальном сценарии. Мы хотели бы отметить, что нашей основной целью было оценить решение, основанное только на зрении, в городских условиях. Перезапуски происходили преимущественно при надежном приеме сигнала GPS (C, D и E на рис. 8(a)). В большинстве регионов алгоритм использовал визуальную информацию в

богатой структурами среды (A и B на рис. 8(a)), что позволяет преодолеть 15*,*7 км маршрута в сложной городской среде. Поэтому мы считаем, что система, переключаемая между визуальным локатором и GPS, может стать реальным дополнительным решением в городских условиях.

1. ВЫВОДЫ

Мы предложили облегченный алгоритм локализации стереокамеры, примененный к ранее полученной 3D-карте. Используя начальное предположение из отслеживания, мы оценили полную позу камеры 6 DOF в пределах карты путем минимизации остаточной глубины. Результаты, полученные на различных наборах данных, показывают, что наш метод применим к различным платформам, включая общедоступную KITTI и наш собственный набор данных. Предложенный подход может стать дополнительным решением для GPS на уровне потребителя, особенно на узких улицах в сложных городских районах, где GPS работает крайне спорадически.

В будущем можно усовершенствовать этот легкий локализатор, интегрировав его с дополнительными датчиками, такими как GPS и инерциальный измерительный блок (IMU). Следует рассмотреть возможность дальнейшего улучшения инвариантности к изменению освещенности и устойчивости к динамическим объектам.

ПОДТВЕРЖДЕНИЕ

Данная работа выполнена при поддержке гранта Корейского MOTIE (№10051867) и проекта [High-Definition Map Based Precise Vehicle Localization Using Cameras and LIDARs], финансируемого корпорацией Naver Labs.

ССЫЛКИ

1. H. Карвальо, П. Дель Мораль, А. Монин и Г. Салют, "Оптимальная нелинейная фильтрация при интеграции GPS/INS", *IEEE Trans. Aerospace and Electronic Sys.* , vol. 33, no. 3, pp. 835-850, 1997.
2. A. Mohamed and K. Schwarz, "Adaptive kalman filtering for INS/GPS," *Journal of geodesy*, vol. 73, no. 4, pp. 193-203, 1999.
3. H. Qi и J. B. Moore, "Подход прямой фильтрации Калмана для интеграции GPS/INS", *IEEE Trans. Aerospace and Electronic Sys.* , том 38, № 2, стр. 687-693, 2002.
4. A. Nu¨chter, K. Lingemann, J. Hertzberg, and H. Surmann, "6D SLAM-3D mapping outdoor environments," *J. of Field Robot.* , том 24, № 8-9, стр. 699-722, 2007.
5. M. Боссе, Р. Злот и П. Флик, "Zebedee: Разработка пружинного 3-мерного датчика дальности с применением для мобильного картографирования", *IEEE Trans. Robot.* , vol. 28, no. 5, pp. 1104-1119, 2012.
6. J. Чжан и С. Сингх, "LOAM: Лидарная одометрия и картирование в реальном времени", в *Proc. Robot..: Science & Sys. Conf.* , vol. 2, 2014.
7. T. Suzuki, M. Kitamura, Y. Amano, and T. Hashizume, "6-DOF локализация для мобильного робота с использованием наружных 3D карт вокселей," in *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.* , 2010, pp. 5737-5743.
8. K. Yoneda, H. Tehrani, T. Ogawa, N. Hukuyama, and S. Mita, "Lidar scan feature for localization with highly precise 3-D map," in *Proc. IEEE Intell. Vehicle Symposium*, 2014, pp. 1345-1350.
9. P. Ruchti, B. Steder, M. Ruhnke, and W. Burgard, "Localization on openstreetmap data using a 3D laser scanner," in *Proc. IEEE Intl. Conf. on Robot. and Automat.* , 2015, pp. 5260-5265.
10. A. Сегал, Д. Хэнел и С. Трун, "Обобщенный МКП", в *Proc. Robot: Science & Sys. Conf.* , 2009.
11. A. Д. Стюарт и П. Ньюман, "LAPS-локализация с использованием внешнего вида предшествующей структуры: 6-дофокусная локализация монокулярной камеры с использованием предшествующих облаков точек", в *Proc. IEEE Intl. Conf. on Robot. and Automat.* , 2012, pp. 2625-2632.
12. R. У. Уолкотт и Р. М. Юстис, "Визуальная локализация на картах LIDAR для автоматизированного городского вождения", в *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.* , 2014, pp. 176-183.

**5832**

перезапустить

20



5.335 6.7793

ABCDE 15

трансляционная ошибка (м)

1000

800

600

400

y (м)

10

5

0

010

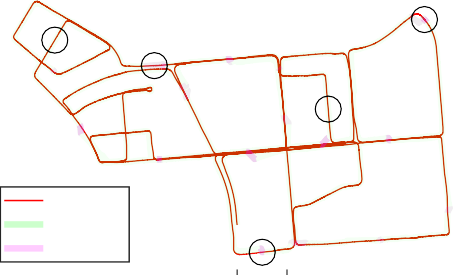
2030

405060

7080

90100

200



E

A

C

B

GT

Бутафорский перезапуск

D

20

боковые вертикальные продольные

0

15

трансляционная ошибка (м)

-200

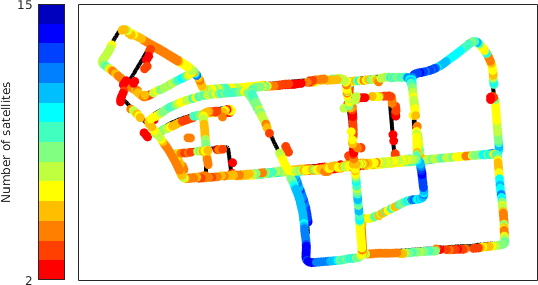
-1000 -800 -600 -400 -2000200400600800 100010

x (м)

1. Траектории5

0

010



0

2030

405060

7080

90100

10

4.9476 3.5102

8

погрешность вращения ( °)

6

4

2

0

010

203040

5060

7080

90100

1. Количество доступных спутников(c) Ошибки локализации

Рис. 8. Результат на нашем наборе данных. (a) показана траектория движения, соответствующая истине, и траектория, оцененная нашим алгоритмом локализации. Образцы изображений выбранных регионов на траектории приведены в верхнем ряду. Количество доступных спутников на маршруте приведено в (b). Три графика в (c) демонстрируют ошибки локализации в процентах от траектории. Первый график показывает трансляционную ошибку, которую можно разделить на трансляционные ошибки в боковом, вертикальном и продольном направлениях, как на втором графике. Ошибка вращения показана на третьем графике. Пурпурная полоса над графиками представляет каждый рестарт.

1. T. Казелиц, Б. Штедер, М. Рунке и В. Бургард, "Локализация монокулярной камеры на трехмерных лидарных картах", в *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.* , 2016, pp. 1926-1931.
2. P. Neubert, S. Schubert, and P. Protzel, "Sampling-based methods for visual navigation in 3D maps by synthesizing depth images," in *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.* , 2017, pp. 2492-2498.
3. Y. Xu, V. John, S. Mita, H. Tehrani, K. Ishimaru, and S. Nishino, "3D point cloud map based vehicle localization using stereo camera," in *Proc. IEEE Intell. Vehicle Symposium*, 2017, pp. 487-492.
4. C. Forster, M. Pizzoli, and D. Scaramuzza, "Air-ground localization and map augmentation using monocular dense reconstruction," in *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.* , 2013, pp. 3971- 3978.
5. J. Энгель, Т. Ш. и Д. Кремерс, "LSD-SLAM: крупномасштабный прямой монокулярный SLAM", в *Proc. European Conf. on Comput. Vision*, 2014, pp. 834-849.
6. D. Мигер, "Геометрическое моделирование с использованием восьмеричного кодирования", *Компьютерная графика и обработка изображений*, том 19, № 2, стр. 129-147, 1982.
7. R. Б. Русу и С. Кузинс, "3D уже здесь: Библиотека облаков точек (PCL)", в *Proc. IEEE Intl. Conf. on Robot. and Automat.* , 2011, pp. 1-4.
8. Г. Брадски и А. Калер, *Learning OpenCV: компьютерное зрение с библиотекой OpenCV*, 2008.
9. H. Хиршмюллер, "Точная и эффективная обработка стереоизображений с помощью полупрограммного метода

глобальное соответствие и взаимная информация", *IEEE Trans. Pattern Anal- ysis and Machine Intell.* , vol. 30, no. 2, pp. 328-341, 2008.

1. G. Sibley, L. Matthies, and G. Sukhatme, "Bias reduction and filter convergence for long range stereo," *Robot. Res.* , pp. 285-294, 2007.
2. R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: универсальная и точная монокулярная система слэма", *IEEE Trans. Robot.* , vol. 31, no. 5, pp. 1147-1163, 2015.
3. R. Куэммерле, Г. Гризетти, Х. Страсдат, К. Конолиге и В. Бургард, "g 2 o: Общая структура для оптимизации графов", в *Proc. IEEE Intl. Conf. on Robot. and Automat.* , 2011, pp. 3607-3613.
4. A. Гейгер, П. Ленц и Р. Уртасун, "Готовы ли мы к автономному вождению? набор эталонных образцов зрения KITTI", в *Proc. IEEE Conf. on Comput. Vision and Pattern Recog.* , 2012, pp. 3354-3361.
5. J. Jeong, Y. Cho, Y.-S. Shin, H. Roh, and A. Kim, "Complex urban lidar data set," in *Proc. IEEE Intl. Conf. on Robot. and Automat.* , Brisbane, May. 2018, в печати.
6. T. Scott, A. A. Morye, P. Pinie´s, L. M. Paz, I. Posner, and P. Newman, "Exploiting known unknowns: Перекрестная калибровка систем лидар-стерео, вызванная сценой", в *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.* , 2015, pp. 3647-3653.
7. A. Д. Стюарт, "Локализация с использованием внешнего вида предшествующей структуры", докторская диссертация, Оксфордский университет, 2014.

**5833**