A. B. Tristanov, O. O. Lukovenkova
Institute of Space Research and Radio Wave Propagation Feb RAS

D. M. Klionsky
Saint Petersburg Electrotechnical University «LETI»

ADAPTIVE APPROACH TO SIGNAL ANALYSIS
(THROUGH THE EXAMPLE OF GEOACOUSTIC EMISSION)

Introduces the results obtained by the authors during the development of an integrated approach to the analysis of one-dimensional geophysical signals. The approach is illustrated on geoacoustic signals. We illustrate the suggested approach using geoacoustic emission signals consisting of a sequence of relaxation impulses with a short front and long decay period. We discuss main approaches to constructing the feature space based on sparse representation of signal segments. Sparse representations are constructed using classical matching pursuit and its adaptive modification based on applying optimization techniques for iterative specification of the parameters of functions obtained as a result of signal decomposition. We provide the results of statistical analysis of decomposition results for real geoacoustic signals. We suggest an approach to segment classification based on extracted features.

Geoacoustic emission, signal data mining, sparse approximation

УДК-519.688

Ар. Ю. Филатов, Ан. Ю. Филатов, К. В. Кринкин
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Б. Чен, Д. Молодан
Массачусетский технологический институт

Методы сравнения качества 2D-SLAM-алгоритмов

Представлен подход для сравнения алгоритмов SLAM, который позволяет найти наиболее точный из них. Акцент исследований сделан на 2D-SLAM-алгоритмах, рассматриваются также 2D-карты, строящиеся после выполнения алгоритма. Предложены три показателя для оценки карт. Цель данной статьи – представить методы количественной оценки качества 2D лазерного SLAM-алгоритма. Объектом рассмотрения служит результирующая карта, и по ее визуальной оценке можно выяснить, какой алгоритм создает лучшую карту, т. е. определить, какая карта содержит наименьшее количество шума, наиболее точно определяет стены, и имеет самое низкое количество артефактов и т. д. Представлены некоторые показатели количественной оценки, а также оценивающий карты фреймворк. Карты, построенные каждым рассмотренным алгоритмом, сравниваются с использованием нескольких показателей, и для каждого запуска назначается оценка. Таким образом, можно оценивать алгоритмы SLAM, даже если истинная карта или траектория не предусмотрены для рассматриваемых наборов данных.

Сравнение SLAM, карта, траектория, gmapping, Cartographer

SLAM определяет часть проблемы навигации для мобильных платформ, в которую входят сбор данных с датчиков, а затем одновременное создание карты и определение местоположения на ней. Задачи, включающие проблему SLAM, встречаются часто, и для их решения в настоящее время используется огромное количество алгоритмов. Они различаются по типам датчиков, способам хранения данных, математическому устройству для обработки данных сканирования и т. д. Таким образом, появляется проблема сравнения доступных алгоритмов и выбора наиболее подходящего.

Чтобы оценить качество алгоритма SLAM, необходимо оценить точность его результатов,

которые могут быть представлены картой и траекторией мобильной платформы, построенными во время прогона. Самый простой способ сравнить результаты нескольких алгоритмов с одной и той же последовательностью данных – рассчитать разницу между построенной и истинной картами. Тем не менее истинные карты зачастую недоступны для многих наборов данных. Для этих последовательностей данных сложно извлечь карту, которую можно было бы сравнить с построенной алгоритмом SLAM. Если набор данных не имеет истинной карты или траектории, необходимо оценить качество результатов алгоритма другими способами. По мнению авторов, таких методов существует немного.

Свойства алгоритмов (скорость работы, разрешение карты и т. п.) не учитывались, поскольку основная цель заключалась в оценке точности вывода. Таким образом, все алгоритмы имели достаточно ресурсов и были настроены на одинаковое разрешение карты.

Современное состояние проблемы. Оценка алгоритмов SLAM всегда была важной, но сложной задачей. Наиболее тривиальный и точный способ количественной оценки оценочной карты, созданной алгоритмом SLAM, заключается в использовании меры расстояния до истинной карты [1]. Это оказывается наилучшим способом оценки построенной карты в принципе, за исключением того факта, что на практике построенные карты могут отличаться друг от друга при нескольких запусках алгоритма. При различных критериях расстояния показатели оценки могут способствовать выявлению более четкой карты.

В этом направлении также проводятся некоторые дополнительные исследования. В [2] обсуждаются недостатки использования глобальной системы отсчета в оценке (например, ошибка штрафа в начале траектории больше, чем в конце) и предлагается альтернативное решение, которое инвариантно к выбору системы отсчета и использует только относительную информацию. В [3], чтобы упростить оценку построенный карты, структура оценки для алгоритмов SLAM строится с использованием истинной траектории и данных лазерного дальномера (генерируется «псевдоистинная карта»). Сгенерированные карты не идеальны, но этот метод также успешно получает карты в реальных жизненных ситуациях с приемлемой точностью.

Однако на практике даже существование истинной траектории не всегда гарантируется. Хотя в [4] предлагается количественный способ сравнения алгоритмов SLAM без идеальной истинной карты или траектории, структура оценки по-прежнему требует ручной работы в каждом случае.

Стоит отметить, что кроме точности карт качество алгоритмов SLAM также оценивается с помощью других метрик, включая точность траектории, измеренную глобальной или относительной ошибкой [5], [6], эффективность, устойчивость к шуму или выбросам, а также надежность различных датчиков и наборов данных [7].

Метрики. Если человеку была представлена карта, сгенерированная алгоритмом SLAM, вполне возможно, что он правильно определит качество изображения. Он может выбрать наложенные копии одной комнаты, кривые стены, неожиданные линии в середине комнаты и т. д. Количественная оценка карты может опираться на ту же идею: извлечь все эти особенности каждой ее части и оценить их влияние на всю карту. Здесь описываются несколько показателей, которые могут помочь определить качество сгенерированной карты, не полагаясь на истинную карту или траекторию. Для анализа не следует довольствоваться одним отдельным показателем, поскольку он может показывать только приблизительные результаты при конкретных условиях. Таким образом, поскольку для анализа используются несколько метрик, можно сделать более точный вывод.

В последующих описаниях формат обрабатываемых карт представляет собой 2D-матрицу – сетку занятости, где каждой ячейке присваивается вероятность того, что она занята. Свободные ячейки в представлении изображения определяются как белые, а чем выше вероятность того, что ячейка занята, тем темнее она на изображении. По умолчанию цвет неизвестной ячейки в представлении изображения также белый, поскольку это облегчает идентификацию сбоев и артефактов.

Для оценки качества алгоритма SLAM представлены следующие показатели:

– доли занятых и свободных клеток, которые позволяют определить, размыты ли стены на карте, и проверить, есть ли дополнительные стены, которые появились из-за отказа алгоритма;

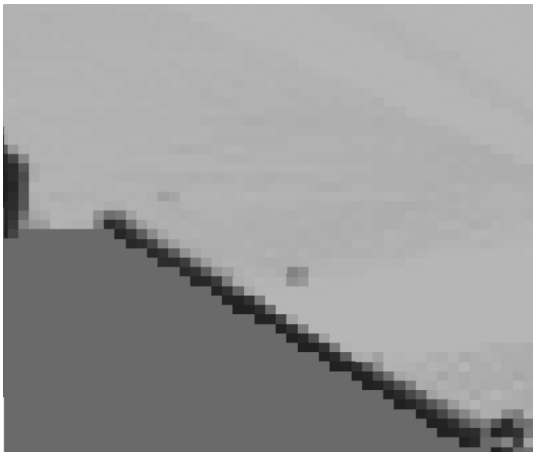
– количество углов, определяющее точность карты. Карта низкого качества должна иметь больше углов, появляющихся в случае перекрытия некоторых частей или проявления неожиданной кривизны прямых стенок;

– количество замкнутых областей, которые представляют следующую идею: если в неиз-

вестной области есть перекрывающиеся области или некоторые артефакты, количество замкнутых областей будет больше, чем на правильно построенной карте.

Доля занятых и свободных клеток. Одна из наиболее заметных особенностей изображения карты – размытость стен. Очевидно, что чем она больше, тем ниже качество карты. На рис. 1 одна и та же стена представлена с высоким (*а*) и низким (*б*) размытием.

а



б

Рис. 1

Чтобы определить размытие, нужно рассчитать доли пикселов, соответствующих стенам и свободному пространству. Определение занятых ячеек может повлечь за собой трудности однозначного выделения стены из свободного пространства.

Самое простое решение – считать любую ячейку с вероятностью больше нуля стеной, но это может привести к появлению артефактов в свободном пространстве. Следующая идея – определение порогового значения для выделения свободных ячеек из занятых. Порог не должен быть жестко запрограммирован и должен быть настраиваемым для каждой карты, так как он может зависеть от наибольшей вероятности ячеек, количества размытости на стене и т. д. Для определения порога предлагается использовать среднее значение всех ячеек. Все клетки, вероятность занятия которых ниже этого порога, должны считаться свободными, а остальные – классифицироваться как занятые.

Доля занятых клеток соответствует качеству карты: чем выше это значение, тем ниже качество алгоритма. Данная метрика позволяет пользователю узнать, существует ли размытие стены. Другое применение – обнаружение дублирования стены на карте. Если на ней присутствуют две очень близкие параллельные линии, можно утверждать, что скорее всего это одна и та же линия, и доля занятых клеток позволяет определить, появились ли на карте ошибки данного типа.

Эта идея работает правильно в предположении, что построенная карта близка к истинной, т. е. построенная и истинная карты должны иметь сходную структуру. Если это предположение не выполняется, использование данной метрики может привести к неправильным результатам. Например, алгоритм теряет данные и создает неполную карту, и в этом случае доля занятых ячеек может быть очень малой. По этой причине данный показатель не может применяться независимо от других рассматриваемых метрик и должен использоваться только как часть комплексного анализа.

Количество углов. Можно утверждать, что чем больше углов выявлено на построенной карте при тестировании на одних данных, тем выше вероятность того, что она менее согласована и имеет больше артефактов. Если считать, что никакая информация не пропущена или не отброшена никаким алгоритмом и все углы будут успешно отражены на карте, то все их можно считать ошибками: артефактами, удвоенными стенами, вызванными несоответствием траектории, и т. д. Разумеется, разный уровень детализации между различными алгоритмами SLAM сделает это сравнение менее убедительным, но, применяя более сильное правило для определения угла, эта метрика может успешно выявить более корректную карту.

В данном случае принимаются во внимание только структурные углы и игнорируются мелкие точки, рассеянные по всей свободной зоне. Эти точки могут быть небольшими препятствиями или просто шумом и, игнорируя их, можно получить более четкий и точный результат.

Чтобы извлечь углы из необработанных карт, значения пикселов карты сначала переназначаются с использованием правила, что чем выше значение ячейки, тем выше вероятность того, что ячейка не занята (в то время как неизвестная область должна быть равна нулю). Затем применяется фильтр Гаусса–Лапласа для извлечения абстрактной структуры новой карты. Основываясь на абстрактной структуре, после удаления небольших дискретных точек (или групп точек) к предварительно обработанной структуре карты применяется угловой детектор Harris [8], чтобы получить на карте структурные углы. Количество углов подсчитывается и считается индикатором качества карты. Эта метрика состоятельна, если сопоставляемые карты, по меньшей мере, одинакового качества.

Замкнутые области. Еще одна характеристика, которая может быть использована для определения качества карты – это количество замкнутых областей на карте. Замкнутая область – это область на карте, полностью ограниченная клетками, обозначенными как занятые или те, наличие в которых препятствия еще не установлено. Такие граничащие клетки можно рассматривать как стены, окружающие открытое пространство.

Существует несколько ситуаций, когда наличие таких областей указывает на сбой. Например, когда комната сканируется несколько раз, но не распознается, так что окончательная карта состоит из небольших поворотов комнаты, наложенных друг на друга (на внешних краях исследуемой области есть несколько замкнутых треугольников). Другая ситуация – отказ процедуры замыкания цикла, когда робот не может распознать, что он вернулся в то же место, где он был запущен. В этом случае можно увидеть визуальное совпадение первой и последней частей построенной карты, которые оказались разнесены в пространстве.

Чтобы получить полезную карту, нужно разделить все ячейки на две категории: занятые/неопределенные и свободные. Значения пикселов неопределенных областей приравниваются к занятым. Затем карта сводится к двоичному изображению с использованием метода Отсу [9]. Далее замкнутые области обнаруживаются с помощью алгоритма Сузуки [9], и процесс повторяется сначала, изменив значение неопределенных ячеек, уменьшая их вероятности быть занятыми. Таким образом выполняется итерация по всем возможным значениям неопределенных областей и возвращается максимальное количество обнаруженных замкнутых областей. Пример извлеченных закрытых областей показан на рис. 2.

Рис. 2

Тестирующий сервис. *Описание решения.* Упомянутые методы оценки реализуются в сервисе, созданном авторами. Этот сервис позволяет пользователю выполнять несколько алгоритмов SLAM на загруженных последовательностях данных и дает количественные и качественные оценки результата.

Набор алгоритмов SLAM был выбран авторами и будет расширяться в будущем. Текущий набор включает gmapping [10], реализацию tinySLAM [11], vinySLAM [12], hectorSLAM [13] и алгоритм Cartograpger от Google [14]. Этот список включает в себя точные алгоритмы, выступающие в роли рефери – gmapping и hectorSLAM, один новый алгоритм – Cartograpger, и алгоритмы из SLAM-конструктора для ROS [15].

Для оценки алгоритмов используются их реализации ROS [16]. Эта операционная система была выбрана потому, что она представляет собой единый способ запуска различных алгоритмов. Более того, она популярна для современных алгоритмов SLAM, и многие из них входят в ROS API. ROS также позволяет предоставлять данные и собирать результаты в едином структурированном формате.

В качестве входных данных в настоящее время используются последовательности MIT [3]. Этот набор данных имеет истинную траекторию, поэтому можно количественно оценивать значение СКО результирующей траектории от истинной. Далее будут добавляться другие наборы данных, например Deutsches Museum [6] или Willow Garage [17]. К сожалению, упомянутые наборы не снабжены истинными траекториями или картами, что уменьшает количество методов, которые могут применяться для оценки алгоритмов SLAM [18]. Наборы данных часто представлены в разных форматах, поэтому существует еще одна задача – преобразование их в единую структуру.

Для запуска тестирующего сервиса требуется, чтобы пользователь выбирал алгоритм SLAM (или несколько алгоритмов), последовательность (или последовательности) и количество итераций, для которых будут выполняться алгоритмы. Тестирующий сервис позволяет настраивать переменные для алгоритмов SLAM. Могут существовать переменные для процесса сопоставления сканирования, для отображения карты и т. д. Таким образом можно оценивать подходы SLAM в аналогичных условиях.

Каждый запуск выполняется на удаленном сервере, чтобы не занимать пользовательские ресурсы. Итерации, в свою очередь, разделяются на параллельные запуски. Для каждого запуска требуются все серверные ресурсы, и следующее выполнение может быть запущено только после завершения всех предыдущих.

Для каждого алгоритма SLAM и каждой последовательности создаются следующие выходные файлы:

– текстовый файл со средним значением СКО-траектории в метрах;

– PGM-файл карты в каждый момент времени (включая GIF-файл как объединение всех этих изображений – эволюцию появления карты);

– текстовый файл с количеством углов, извлеченных из карты;

– текстовый файл с долей занятых ячеек относительно всех ячеек;

– текстовый файл с количеством замкнутых областей.

Далее утилита извлекает информацию из всех этих файлов и отображает ее как html-страницу. Она представляет собой таблицу со значениями СКО только в том случае, если последовательность данных снабжена истинной траекторией. Эта таблица легко анализируется, потому что меньшее значение СКО означает, что соответствующий алгоритм SLAM имеет лучшие результаты. Далее приводится пример таблицы, заполненной значениями СКО-траекторий, которые получены из ранее упомянутых алгоритмов SLAM.

Другая таблица, генерируемая утилитой, состоит из набора карт – изображений, которые иллюстрируют результирующие представления среды, полученные после запуска SLAM. Эта оценка – не количественная, но позволяет пользователю распознавать случаи, когда один алгоритм SLAM отрабатывает лучше, чем другой. Например, алгоритм может отлично работать в коридоре, но ошибаться, если предоставляются данные, включающие в себя огромные ошибки измерения. Примеры карт представлены на рис. 3 (*а* – Cartographer; *б* – gmapping; *в* – hectorSLAM; *г* – tinySLAM; *д* – vinySLAM). Кроме

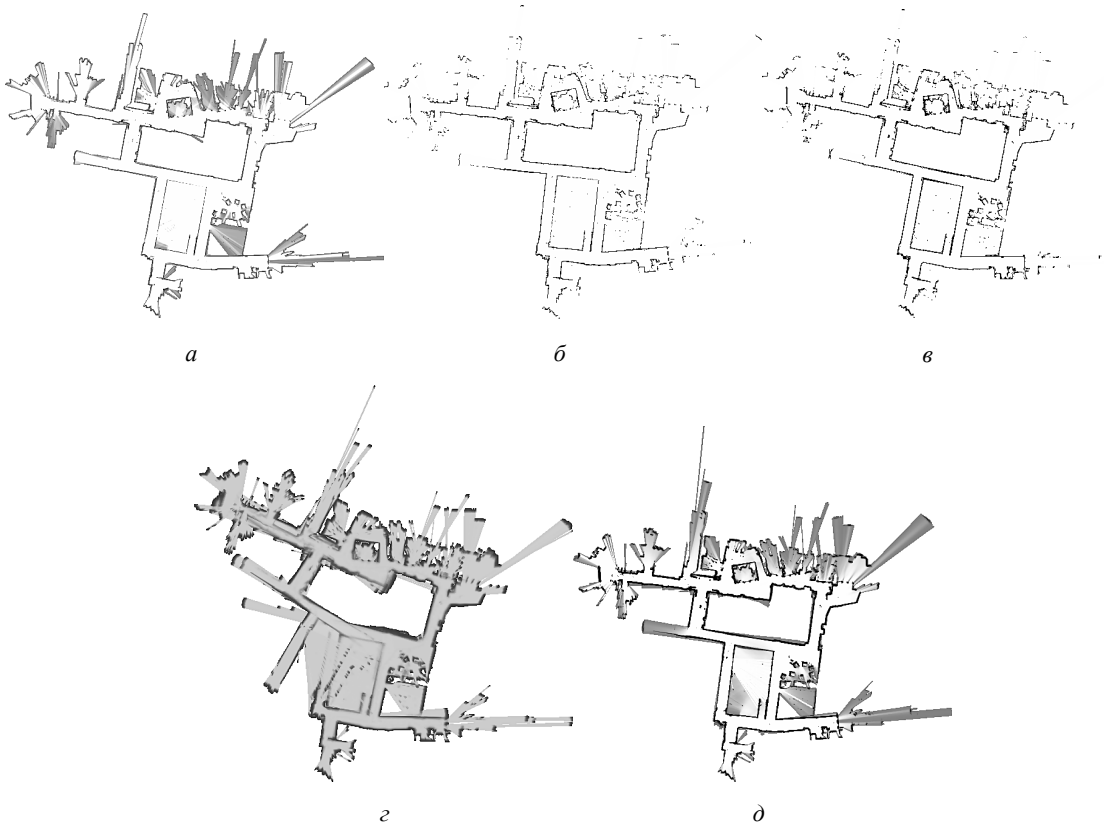


Рис. 3

Последовательность	Длина, м	SLAM-алгоритм				
		tinySLAM	vinySLAM	gmapping	Cartographer	hectorSLAM
2011-01-19-07-49-38	68	2.432 ± 0.488	1.581 ± 0.579	0.239 ± 0.011	0.275 ± 0.021	0.211 ± 0.007
2011-01-20-07-18-45	76	0.434 ± 0.061	0.187 ± 0.012	0.243 ± 0.044	0.208 ± 0.021	0.226 ± 0.004
2011-01-21-09-01-36	87	0.235 ± 0.006	0.151 ± 0.027	0.205 ± 0.007	0.482 ± 0.059	0.207 ± 0.003
2011-01-24-06-18-27	87	0.266 ± 0.013	0.162 ± 0.023	0.256 ± 0.043	0.202 ± 0.0022	0.228 ± 0.004
2011-01-25-06-29-26	109	0.249 ± 0.011	0.106 ± 0.019	0.231 ± 0.011	0.515 ± 0.039	0.236 ± 0.003
2011-01-28-06-37-23	145	2.064 ± 0.0282	0.569 ± 0.444	0.374 ± 0.038	0.628 ± 0.028	0.311 ± 0.004
2011-03-11-06-48-23	245	0.660 ± 0.118	0.667 ± 0.231	0.496 ± 0.022	1.336 ± 0.086	6.611 ± 2.455

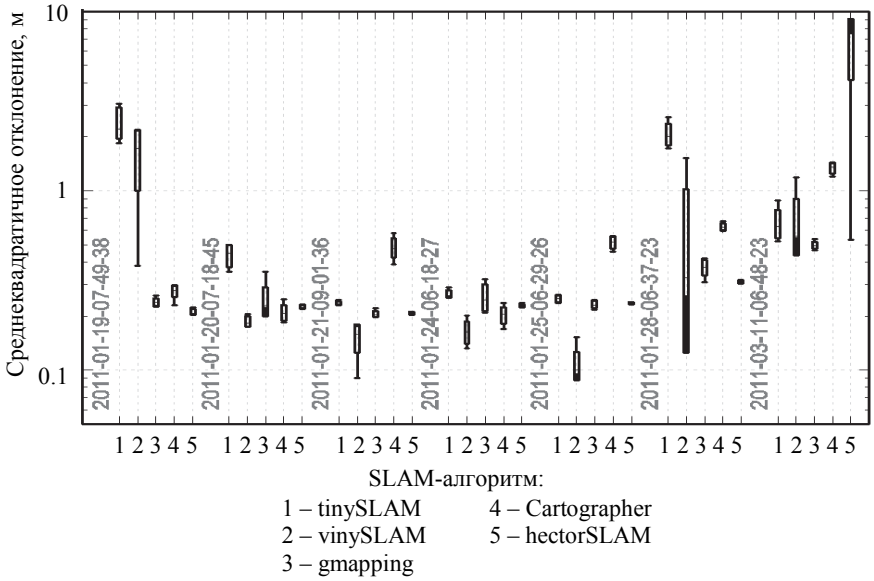


Рис. 4

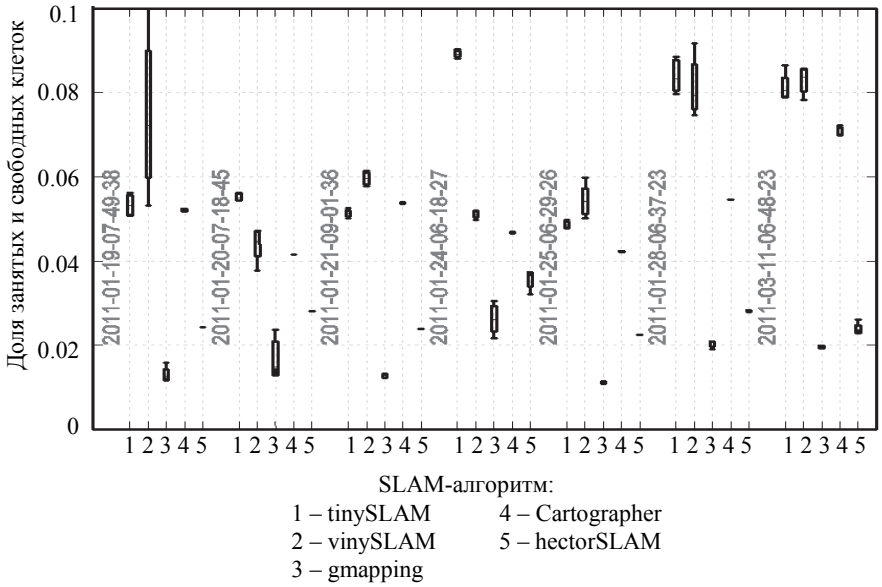


Рис. 5

того, есть еще три показателя с количественными результатами, в которых представлено количество выделенных углов, замкнутых областей и отношение занятых ячеек к общему количеству клеток на картах. Эти показатели – относительные и могут применяться для оценки, только когда запускаются несколько алгоритмов SLAM в одной и той же последовательности данных. С другой стороны, эти метрики не требуют истинной карты или траектории. Значения метрик представлены на рис. 4–7. По оси абсцисс на графиках этих рисунков располагаются алгоритмы SLAM

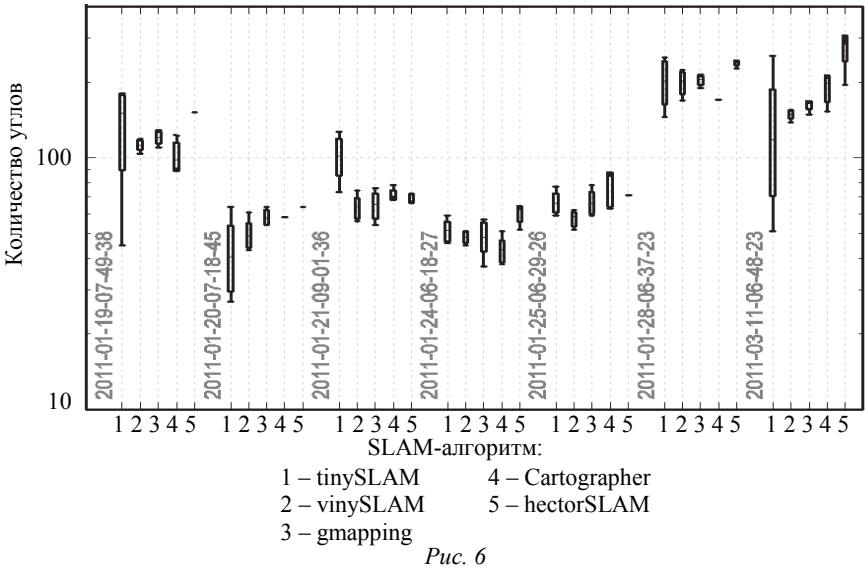


Рис. 6

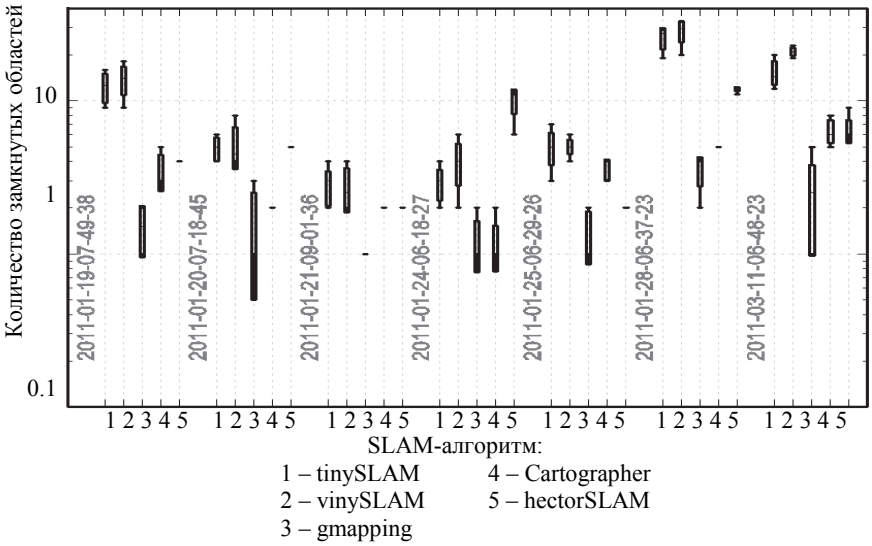


Рис. 7

сравниваемых типов (*t* – tinySLAM, *v* – vinySLAM, *g* – gmapping, *c* – Cartographer, *h* – hectorSLAM), сгруппированные по последовательностям данных (2011-01-19-07-49-38, 2011-01-20-07-18-45 и т. д.). По оси ординат располагаются значения рассматриваемых метрик. Сами значения на графиках представляют собой доверительный интервал для каждой метрики, построенный по нескольким результатам запуска алгоритмов на указанных последовательностях данных.

Результаты экспериментов. Результаты эксперимента представлены в таблице и на рис. 3. Была проведена оценка на последовательностях данных MIT [3], которые имеют разную длину и структуру окружающей среды (разное количество комнат и коридоров). В таблице и на рис. 3 представлены значения среднеквадратичного отклонения траектории. График на этом рисунке имеет логарифмическую шкалу ординат для лучшего визуального отображения. Результаты показывают, что gmapping, Cartographer и hectorSLAM наиболее надежны (имеют самую низкую дисперсию), но, например, hectorSLAM или Cartographer могут потерпеть неудачу на некоторых последовательностях и создать несогласованную карту. Пример такого исполнения представлен на рис. 8.

Доля занятых клеток отражена на рис. 5. Он показывает, что tinySLAM, vinySLAM и Cartographer имеют эффект размытости на стенах, а gmapping и hectorSLAM – нет. Судя по данному описанию этой метрики наиболее точным алгоритмом должен быть gmapping, однако практика показывает, что и другие алгоритмы также могут не уступать ему.

Количество углов на карте для каждого алгоритма представлено на рис. 6. Вертикальная ось имеет логарифмический масштаб. Значение этой метрики показывает, что vinySLAM обычно работает лучше, чем Cartographer и hectorSLAM, и, кроме того, имеет небольшую дисперсию. Это

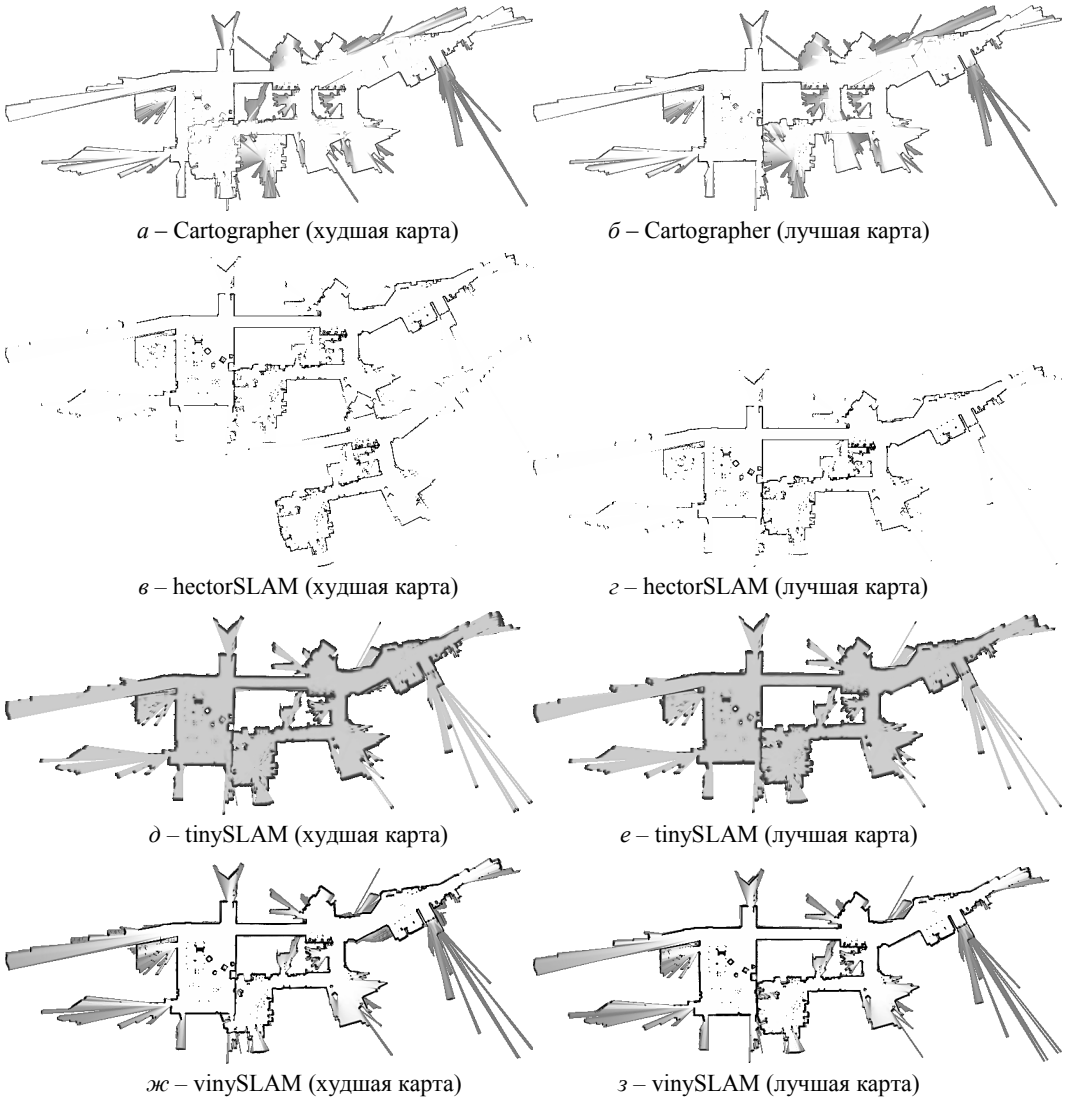


Рис. 8

означает, что карты, созданные vinySLAM, имеют более точные стены и меньше дублируют углы. Кроме того, vinySLAM показывает близкие, а иногда и лучшие результаты относительно gmapping, а значит, это один из самых точных алгоритмов с точки зрения данной метрики. В то же время, tinySLAM обладает наибольшей дисперсией, поскольку использует математический аппарат, основанный на случайных величинах и, следовательно, данный алгоритм не может быть корректно оценен с помощью этой метрики.

Количество замкнутых областей – наиболее чувствительная метрика для случайных алгоритмов, как показано на рис. 7. На рисунке ось ординат также имеет логарифмический масштаб. Огромное преимущество gmapping объясняется небольшой долей занятых ячеек в этом алгоритме: несколько областей, которые можно было считать закрытыми, не учитывались из-за отсутствия границы. С другой стороны, предыдущая метрика также показывает, что gmapping – очень точный алгоритм, и результаты на рис. 7 подтверждают этот факт.

Рассматривая результаты, можно сделать вывод, что gmapping, Cartographer и vinySLAM – наиболее точные из пяти рассмотренных алгоритмов. Gmapping оказывается самым точным, но сложно определить, какой алгоритм должен быть на втором месте. Четвертое место занимает hectorSLAM, в то время как tinySLAM оказывается на последнем месте.

В статье представлены введенные метрики, которые определяют характеристики карт, генерируемых алгоритмами SLAM, и на основе этих характеристик отличают высококачественную от низкокачественной карты одной и той же последовательности. Эти показатели хорошо подходят для реалистичных применений алгоритмов SLAM, где истинные карты и траектории движения не всегда доступны. Кроме того, они дают другую информацию о создании карты, важной в практическом применении, но не имеющей коли-

чественных решений: сигнализируют о появлении возможных артефактов и сбоев при создании карты. Также представлена структура количественной оценки алгоритмов SLAM, применявшаяся для оценки нескольких существующих методов с новыми метриками. Результаты применения вводимых метрик совпали с результатом существующей (вычисление среднеквадратического отклонения) [19], [20], что позволяет использовать их наравне с этой метрикой.

СПИСОК ЛИТЕРАТУРЫ

1. A comparison of SLAM algorithms based on a graph of relations / W. Burgard, C. Stachniss, G. Grisetti, B. Steder, R. Kmmerle, C. Dornhege, M. Ruhnke, A. Kleiner, J. D. Tards // Intern. Conf. on Intelligent Robots and Systems. St. Louis: IEEE Robotics, 2009. P. 2089–2095.
2. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age / C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, J. J. Leonard // IEEE Trans. on Robotics. 2016. № 32 (6). P. 1309–1332.
3. The mit stata center dataset / M. Fallon, H. Johannsson, M. Kaess, J. J. Leonard // Intern. J. of Robotics Research. 2013. № 32 (14). P. 1695–1699.
4. Funke J., Pietzsch T. A framework for evaluating visual slam // Proc. of the British Machine Vision Conf. (BMVC). Washington: BMVA Press, 2009. Vol. 6. P. 156-163.
5. Gerkey B. Ros SLAM gmapping. URL: http://wiki.ros.org/slam_gmapping, 2010 (дата обращения 31.07.2017).
6. 2d cartographer backpack deutsches museum, 2016. URL: https://google-cartographer-ros.readthedocs.io/en/latest/ (дата обращения 31.07.2017).
7. Harris C., Stephens M. A combined corner and edge detector // Alvey vision conf. Manchester: UK, 1988. Vol. 15. P. 5230–5244.
8. Real-time loop closure in 2d lidar SLAM. In Robotics and Automation (ICRA) / W. Hess, D. Kohler, H. Rapp, D. Andor // IEEE Intern. Conf. IEEE, 2016. P. 1271–1278.
9. Huletski A., Kartashov D. A SLAM research framework for ros. // Proc. of the 12th Central and Eastern European Software Engin. Conf. in Russia. M.: ACM, 2016. P. 12.
10. Huletski A., Kartashov D., Krinkin K. Evaluation of the modern visual slam methods // 2015 Artificial Intelligence and Natural Lang. and Inform. Extraction, Social Media and Web Search FRUCT Conf. (AINL-ISMW FRUCT). Yaroslavl: IEEE Robotics, 2015. P. 19–25.
11. Huletski A., Kartashov D., Krinkin K. Tinyslam improvements for indoor navigation // Multisensor Fusion and Integration for Intelligent Systems (MFI), 2016 IEEE Intern. Conf. IEEE, 2016. P. 493–498.
12. Jaulmes R., Molin E., Obriet-Leclef J. Towards a quantitative evaluation of simultaneous localization and mapping methods // Control Architecture of Robots Nat. Conf. Hamburg: IEEE Robotics, 2009. P. 274–284.
13. A flexible and scalable slam system with full 3d motion estimation / S. Kohlbrecher, J. Meyer, O. Stryk, U. Klingauf // Proc. IEEE Intern. Symposium on Safety, Security and Rescue Robotics (SSRR). IEEE, 2011. P. 1454–1461.
14. On measuring the accuracy of SLAM algorithms / R. Kummerle, B. Steder, C. Dornhege, M. Ruhnke // Autonomous Robots. 2009. Vol. 27 (4). P. 387–407.
15. Li H., Zhang Q., Zhao D. Comparison of methods to efficient graph slam under general optimization framework // Automation (YAC) 32nd Youth Acad. Ann. Conf. of Chinese Assoc. IEEE. 2017. P. 321–326.
16. Mason J., Marthi B. An object-based semantic world model for long-term change detection and semantic querying // Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ Intern. Conf. IEEE. 2012. P. 3851–3858.
17. O'Kane J. M. A gentle introduction to ros, 2014. URL: https://www.cse.sc.edu/~jokane/agitr/agitr-letter.pdf (дата обращения 31.07.2017).
18. Otsu N. A threshold selection method from gray-level histograms // IEEE Trans. on Systems, Man, and Cybernetics. 1979. № 9 (1). P. 62–66.
19. Santos J. M., Portugal D., Rocha R. P. An evaluation of 2d SLAM techniques available in robot operating system // 2013 IEEE Intern. Symposium on Safety, Security, and Rescue Robotics (SSRR). Linkoping: IEEE Robotics, 2013. P. 1–6.
20. Suzuki S., Abe K. Topological structural analysis of digitized binary images by border following // Comp. Vision, Graphics, and Image Proc. 1985. № 30(1). P. 32–46.

Ar. Yu. Filatov, An. Yu. Filatov, K. V. Krinkin

Saint Petersburg Electrotechnical University «LETI»

B. Chen, D. Molodan

Massachusetts Institute of Technology

METHODS OF COMPARING THE QUALITY OF 2D-SLAM-ALGORITHMS

Presents an approach for comparing SLAM algorithms, which allows you to find the most accurate. The emphasis of the research is made on 2D-SLAM-algorithms, and 2D maps are considered, which is built after the algorithm is executed. This article presents three indicators for the evaluation of maps. The purpose of this article is to present methods for quantifying the quality of the 2D laser SLAM algorithm. The object of consideration is the resulting map, and by its visual evaluation it is possible to find out which algorithm creates the best map, i. e., to determine which map contains the least amount of noise, the most accurate walls, the lowest number of artifacts, etc. This article presents some indicators of quantification. The article also presents a map estimating framework. Maps, which are built by each algorithm considered, are compared with the use of several indicators, and for each launch an evaluation is assigned. Thus, it is possible to evaluate SLAM algorithms, even if the true map or trajectory is not provided for the data sets under consideration.

SLAM comparison, map, trajectory, gmapping, Cartographer