

Хусайнов Наиль Шавкятович – e-mail: naile.khussainov@gmail.com; кафедра математического обеспечения и применения ЭВМ; зав. кафедрой; к.т.н.; доцент.

Щербинин Виктор Викторович – "ЦНИИАГ", г. Москва; 127018, Москва, ул. Советской Армии, 5; тел.: 84956006317; д.т.н.; начальник научно-технического отделения.

Kravchenko Pavel Pavlovich – Southern Federal University; e-mail: Pavelkravch@gmail.com; 44, Nekrasovsky lane, Taganrog, 347928, Russia; phone: +78634393545; the department of software engineering; dr. of eng. sc.; professor.

Khusainov Nail Shavkyatovich – e-mail: naile.khussainov@gmail.com; the department of software engineering; head of department; dr. of eng. sc.; associate professor.

Scherbinin Victor Victorovich – Central Research Institute of Automatics and Hydraulics; e-mail: mail_dv@mail.com; 127018, Moscow, 5, Sovetskaya Army street, phone: +74956006317; dr. of eng. sc.; head of research department.

УДК 007:621.865.8

DOI 10.18522/2311-3103-2017-1-172184

В.Н. Казьмин**ВЕРОЯТНОСТНЫЙ ПОДХОД К РЕШЕНИЮ ЗАДАЧИ SLAM
В ТРЁХМЕРНОМ ПРОСТРАНСТВЕ***

Предложен и рассмотрен подход к решению центральных для автономной робототехники задач определения координат подвижных роботизированных объектов и построения модели окружающего пространства по данным бортовой системы технического зрения, основанный на применении численного метода Монте-Карло для задачи локализации. Суть метода состоит в применении многочастичного фильтра для последовательного вычисления плотности распределения вероятностей, которая задаёт возможные положения робота в пространстве. Особенностью реализации метода является использование выделенных в дальнометрическом кадре плоскостей для измерения положения робота относительно карты, путём сопоставления их с уже имеющимися в базе данных и решения системы уравнений, построенной на параметрах найденных пар соответствующих друг другу плоскостей. При навигации, помимо геометрии пространства, используется яркостная раскраска выделенных плоскостей с целью повышения точности и разрешения геометрически неопределённых ситуаций. После определения положения робота найденные в кадре плоскости вносятся в базу данных с объединением и усреднением с уже имеющейся в ней информацией и используются для навигации в последующих циклах алгоритма. Приведены результаты работы созданных алгоритмов и программ, решающих поставленные задачи в темпе движения робота; осуществлена оценка ошибок и производительности. На основе анализа результатов теоретических и экспериментальных исследований сделано заключение, что предлагаемый подход обеспечивает переход от больших объемов исходной информации к компактным описаниям внешней среды, содержащим навигационные данные, что позволяет эффективно решать задачи локализации и автономного управления движением мобильных роботов и беспилотных летательных аппаратов.

Мобильный робот; беспилотный летательный аппарат; система технического зрения; многочастичный фильтр; локализация; реконструкция модели окружающего пространства; метод Монте-Карло.

* Работа выполнена при поддержке гранта РФФИ №16-29-04178 офи_м.

V.N. Kazmin

PROBABILISTIC APPROACH IN 3D SLAM PROBLEM

Approach for solving central in autonomous robotics SLAM problem for moving robotic objects using data of onboard sensors, based on Monte Carlo localization, proposed and considered. In this method the particle filter is used to recursively construct posterior conditional probability density function from previous robot's state and sensors measurement. A feature of the implementation of the method is the use of the detected in rangefinder data planes for measuring the robot position relative to the map, by comparing them with those already in the database and solve a system of equations constructed on the parameters of found pairs corresponding to each other plane. When navigating, in addition to the space geometry, brightness coloring of segmented planes is used in order to improve the accuracy and resolution of geometrically uncertain situations. After determining the position of the robot, founded in the frame planes are entered into the database and averaged with the existing information, and they are used for navigation in the subsequent cycles of the algorithm. The results of the work of created algorithms and software, solving the task in the rate of motion of the robot in the real world, are shown; errors and performance are estimated. Based on the analysis of theoretical and experimental studies it concluded that the proposed approach provides a transition from large amounts of original range-finder data to a compact description of an environment and can effectively solve the problem of autonomous motion control of mobile robots and unmanned aerial vehicles.

Mobile robot; unmanned aerial vehicle; vision system; particle filter; localization; environments model reconstruction; Monte Carlo localization.

Введение. Доля операций и мероприятий, проводимых в индустриально-городской среде, с привлечением роботизированных средств достаточно велика и по оценкам экспертов наблюдается устойчивая тенденция ее роста. Вследствие экранирования в данных условиях канала связи более востребованы здесь не дистанционно-управляемые РТК и БЛА, а – автономные. Центральными задачами при создании автономных систем управления являются задачи навигации и реконструкции окружающего пространства. Для рассматриваемых сред характерно также и экранирование и искажение навигационных полей, поэтому перспективными здесь являются методы экстремальной навигации [1], основанные на обработке дальнометрических изображений, получаемых в процессе движения, тем более, что для реконструкция окружающего пространства используются эти же данные.

В предлагаемом вероятностном подходе к решению задачи навигации учитывается и оценивается мера неопределённости при определении состояния робота и в поступающей с датчиков информации. Положение робота представляется плотностью распределения вероятности в пространстве возможных положений, а при обработке сигналов датчиков учитывается наличие случайных шумов. В вероятностной робототехнике центральной является получившая практическое подтверждение гипотеза, что система управления, при проектировании которой учитывается возможная неопределённость состояния робота и поступающей извне информации, функционирует в недетерминированных условиях надёжнее, чем та, которая этого не учитывает. Такой подход позволяет повысить робастность системы и устойчивость работы в непредсказуемых ситуациях, восстанавливать функционирование после возникновения сбоев.

Использование непосредственно исходных облаков точек требует значительных вычислительных ресурсов и памяти. Поэтому актуальной становится задача сжатия исходной информации. Одним из перспективных путей сжатия исходных дальнометрических изображений искусственных сред является их представление в виде совокупности геометрических примитивов (плоскостей, линий, вершин). По сравнению с исходным изображением такие линейные объекты характеризуются значительно меньшей размерностью и зашумленностью, а также возможностью

нахождения различных отношений между ними и перехода к семантическому описанию окружающего пространства [2]. Такой подход значительно снижает объем хранимой информации и позволяет эффективно решать задачи класса SLAM.

Обзор существующих решений. Существует большое количество подходов к задаче одновременной навигации и построения карты (Simultaneous Localization and Map Building - SLAM), различающихся формой представления модели окружающего пространства и способом определения перемещения робота.

Итеративный алгоритм ближайших точек [3] (Iterative Closest Point – ICP) – один из наиболее распространённых методов для определения относительного смещения и ориентации двух пересекающихся облаков точек, часто используемый для задач локализации. На базе этого метода реализованы проекты [4, 5], использующие квадрокоптер для построения карт внутри [4] и вне помещений [5].

В ряде работ для навигации используются плоскости, выделенные в облаках точек. В [6] используется минимум 3 соответствующих друг другу пары пересекающихся плоскостей для вычисления линейного относительного перемещения сенсора. Соответствие определяется с помощью оригинальной стратегии поиска. В [7] для определения взаимного смещения и ориентации, комбинаторно перебираются тройки плоскостей, выделенные в двух соседних дальнометрических изображениях. В обеих работах используются дальнометрические изображения получаемые из стационарных положений.

В [8] реализована система на базе квадрокоптера, использующая сканирующий дальномер для навигации и 3D ToF камеру для построения объёмной модели представленной в виде многослойной сетки занятости пространства.

В [9] используется представление пространства в виде так называемой карты элементов поверхностей разного разрешения (MRSMapy – Multi-Resolution Surfel Maps) [10], в которых используется иерархичное разбиение пространства на ячейки разного размера с помощью структуры типа восьмеричного дерева. Каждая ячейка такой карты представляет геометрию элемента поверхности (Surfel – Surface Element) и некоторые другие её свойства, например, цвет и отражающую способность. Измерения осуществляются с помощью непрерывно вращающегося 2D лазерного дальномера, а для определения 3D положения мобильного робота используется численный метод Монте-Карло для локализации (Monte Carlo Localization – MCL).

В [11] также используется разбиение пространства с помощью структуры типа восьмеричного дерева (octree), но ячейки представляют занятое и свободное с некоторым значением вероятности пространство. В [12] показано использование octree и MCL для определения 3D положение робота гуманоида по данным 2D сканирующего лазерного дальномера.

Предлагаемое решение. Для решения задачи SLAM предлагается использовать численный метод Монте-Карло [13] для задачи локализации, в котором положение робота представляется плотностью распределения вероятности, аппроксимированной набором частиц, каждая из которых несёт информацию о некотором конкретном положении робота. Особенностью реализации метода является использование выделенных в дальнометрическом кадре плоскостей для измерения положения робота относительно карты, путём сопоставления их с уже имеющимися в базе данных и решения системы уравнений, построенной на параметрах найденных пар соответствующих друг другу плоскостей. При навигации, помимо геометрии пространства, предлагается использовать яркостную раскраску выделенных плоскостей с целью повышения точности и разрешения геометрически неопределённых ситуаций. В предлагаемом решении, в отличие от [4, 5], исходное облако точек используется только для выделения геометрических объектов, кото-

рые в дальнейшем используются для навигации. А по сравнению с [6, 7] работа возможна даже при наличии только одной пары плоскостей и не требуется сложных алгоритмов поиска соответствий. Благодаря использованию геометрических объектов, в предлагаемом подходе измерение положения осуществляется быстрее, чем в [8–12].

Для сканирования геометрии пространства используется беспилотный летательный аппарат (БПЛА) с жестко закрепленным малогабаритным 2D-лазерным дальномером. Сенсор установлен на аппарате так, чтобы плоскость сканирования была вертикальна, а получение дополнительной степени сканирования осуществляется за счет управляемого вращения аппарата вокруг вертикальной оси [14], [15]. В случае мобильного робота сканирование может осуществляться за счёт непрерывно вращающегося 2D дальномера, установленного на поворотное устройство.

Принятые обозначения. Обозначения систем координат (СК): body (“b”) – система координат, связанная с корпусом летательного аппарата; map (“m”) – неподвижная СК, в которой строится карта пространства; odom (“o”) – неподвижная СК, относительно которой положение аппарата вычисляется по инерциальным датчикам.

Представление ориентации и положения [16]: a_cR – матрица поворота, задающая ориентацию СК c относительно СК a ; ${}^a_c\bar{t}$ – вектор смещения начала СК c относительно СК a .

Обозначение векторов, точек и преобразований координат: ${}^a\bar{n}$ – вектор, заданный в СК a ; ${}^a\bar{P} = [x \ y \ z]^T$ – точка, заданная в декартовой СК a ; ${}^a\bar{P} = [x \ y \ z \ 1]^T$ – точка, заданная в однородных координатах; ${}^a_c\bar{T} = \begin{bmatrix} {}^a_cR & {}^a_c\bar{t} \\ 0 & 1 \end{bmatrix}$ – преобразование координат для точек заданных в однородных координатах.

Обозначение облаков точек и плоскостей: aO_T – облако точек, заданное в некоторой СК a ; ${}^a\Pi = \{ {}^a\bar{n}, {}^a\mathbf{p}, {}^a_{//}\mathbf{T}, A \}$ – плоскость, заданная нормальным уравнением ${}^a\bar{n} \cdot {}^a\bar{x} - {}^a\mathbf{p} = 0$ в СК a ; ${}^a_{//}\mathbf{T}$ задаёт преобразование из связанной с плоскостью системы координат в СК a ; A – множество составляющих плоскость частей.

Метод Монте-Карло для задачи локализации. В работе используется численный метод Монте-Карло для решения задачи локализации [13], который основан на получении большого числа реализаций стохастического процесса. При вероятностном подходе положение робота представляется плотностью распределения вероятности в пространстве возможных положений. В MCL плотность вероятности $p(x_t | z_{1:t}, u_{1:t})$ аппроксимируется большим набором частиц $X_t = \{x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[N]}\}$, и с помощью многочастичного фильтра (МЧФ) рекурсивно вычисляется из её предыдущего значения $p(x_{t-1} | z_{1:t-1}, u_{1:t-1})$. Каждая n -ая частица представляет собой гипотезу о вероятном состоянии робота в момент времени t . Так как $x_t^{[n]} \sim p(x_t | z_{1:t}, u_{1:t})$, то чем больше частиц сосредоточено в некоторой области пространства состояний, тем больше вероятность, что здесь находится истинное состояние системы.

Алгоритм MCL включает в себя два основных шага: предикцию и коррекцию.

Предикция выполняется по информации, полученной с инерциальных датчиков. Если с предыдущего момента времени $t-1$ робот сместился на вектор \bar{t}_{off} , а

относительный поворот за это время описывается матрицей поворота R_{off} , то, используя модель движения, получим предварительный набор X_t , частицы которого ($\tilde{x}_t^{[n]} = \{ {}^m_b \tilde{R}^{[n]}, {}^m_b \tilde{t}^{[n]} \}$) определяются по формулам:

$$R_\varepsilon = R_z(\varepsilon_{\alpha 1})R_y(\varepsilon_{\alpha 2})R_x(\varepsilon_{\alpha 3})$$

$${}^m_b \tilde{R}^{[n]} = {}^m_b R^{[n]} R_\varepsilon R_{off}$$

$${}^m_b \tilde{t}^{[n]} = {}^m_b t^{[n]} + (1 + \varepsilon_{\alpha 4}) R_\varepsilon \bar{t}_{off}$$

где ε_v – нормально распределённая случайная величина с нулевым средним значением и с дисперсией, равной v .

На шаге коррекции вычисляется искомая плотность распределения вероятности $p(x_t | z_{1:t}, u_{1:t})$, аппроксимированная набором частиц X_t , по предварительному значению $\tilde{p}(x_t | z_{1:t-1}, u_{1:t})$ и полученному в момент времени t измерению z_t .

В [2] было показано, что, зная соответствующие друг другу пары плоскостей карты и текущего кадра (рис. 1), можно определить положение и ориентацию аппарата относительно карты пространства.

Для этого необходимо для M пар плоскостей записать переопределённую в общем случае систему нелинейных уравнений (значение векторов в этой системе раскрыто на рис. 1) и решить ее относительно ${}^m_b \bar{t}$ и ${}^m_b R$, которые являются искомыми перемещением и ориентациями.

$$\begin{cases} {}^b \bar{n}_1 \cdot {}^b p_1 = ({}^m p_1 - {}^m \bar{n}_1 \cdot {}^m \bar{t}_b) ({}^m_b R)^{-1} \cdot {}^m \bar{n}_1, \\ {}^b \bar{n}_2 \cdot {}^b p_2 = ({}^m p_2 - {}^m \bar{n}_2 \cdot {}^m \bar{t}_b) ({}^m_b R)^{-1} \cdot {}^m \bar{n}_2, \\ \dots \\ {}^b \bar{n}_M \cdot {}^b p_M = ({}^m p_M - {}^m \bar{n}_M \cdot {}^m \bar{t}_b) ({}^m_b R)^{-1} \cdot {}^m \bar{n}_M, \end{cases}$$

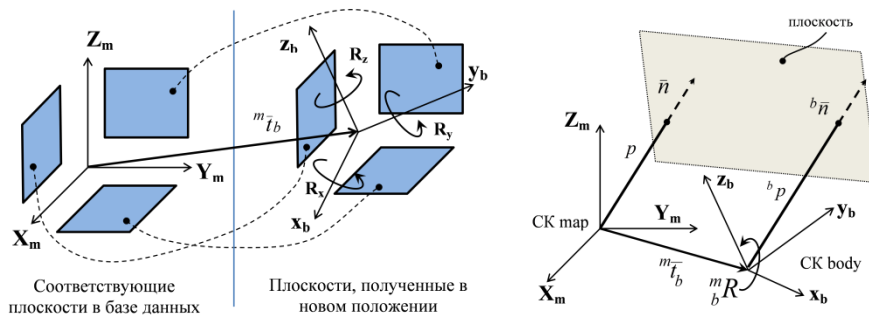


Рис. 1. Иллюстрация метода определения перемещения по соответствующим парам плоскостей

В соответствии с этим, на первом этапе для каждой частицы $\tilde{x}_t^{[n]} = \{ {}^m_b \tilde{R}^{[n]}, {}^m_b \tilde{t}^{[n]} \}$ выполняется поиск парных плоскостей и формируется указанная система уравнений, в результате решения которой (с помощью метода Левенберга-Марквардта) получается некоторое новое значение $\tilde{x}_t^{*[n]}$.

Здесь следует особо отметить, что, в зависимости от текущего участка рабочего пространства, может быть найдено одна, две и более непараллельных друг другу пар плоскостей, и в зависимости от этого геометрическое место частиц $\tilde{\chi}_t^{*[n]}$ будет иметь различную конфигурацию (рис. 2): для 3-х и более пар оно будет стремиться к точке, для 2-х пар – к линии, для 1-ой пары – к плоскости.

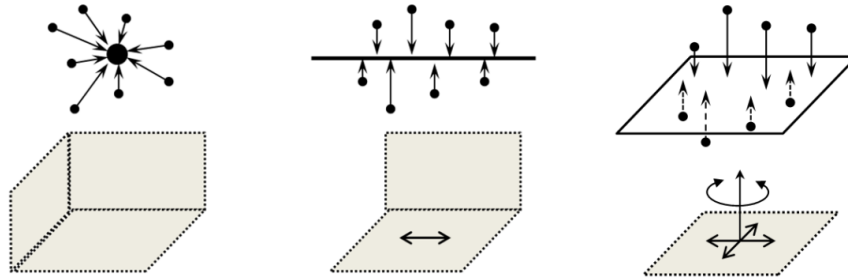


Рис. 2. Геометрическое место частиц при различном количестве пар плоскостей

Для разрешения неопределённости на втором этапе коррекции для каждой частицы вычисляется вес. На рис. 3 показано для сравнения фото участка комнаты и её вид, полученный сбором строк интенсивностей отражённого сигнала лазерного дальномера. В работе [17] было показано, что значения интенсивностей могут быть успешно использованы для повышения точности навигации, и что для получения более стабильного результата необходимо осуществлять поправку принятого сигнала с учётом угла падения луча на поверхность. Здесь это возможно сделать без существенных затрат ресурсов, так как ориентация плоскостей определяется изначально.



Рис. 3. Иллюстрация интенсивностей отражённого сигнала лазерного дальномера: слева – интенсивности отражённого сигнала, справа – фото того же участка комнаты

Для определения веса i -ой частицы в данной работе используется средняя сумма абсолютных разностей (SAD) значений усреднённых интенсивностей, полученная по всем пересекающимся частям всех M пар плоскостей, найденных для данной частицы:

$$\overline{SAD}_i = \frac{1}{N_i} \sum_{j=1}^M \left(\sum_{k \in b_{A_j}^i \cap m_{A_j}} \left| \bar{I}_{k,j}^b - \bar{I}_{k,j}^m \right| \right),$$

где ${}^b A_i^*$ – прореженное множество частей i -ой плоскости в текущем кадре; ${}^m \bar{I}_{k,i}$ – усреднённая интенсивность той части плоскости из карты, в которую проецируется центр ячейки ${}^b a_{k,i}$ текущего кадра; N_i – общее число пересекающихся частей пар плоскостей, полученной для данной частицы (представление плоскостей будет описано ниже).

Значение \overline{SAD} имеет минимум в точке наибольшего соответствия распределения интенсивностей, а для весов необходимо иметь максимум в этой же точке. Поэтому выполняется инвертирование: $\overline{SAD}_i^* = \max_i(\overline{SAD}_i) - \overline{SAD}_i$. В итоге нормированный вес частицы определяется по формуле: $w_i = \overline{SAD}_i^* / \sum_i \overline{SAD}_i^*$

Заключительным этапом коррекции является выборка по значимости, целью которого является формирование набора частиц X_i , в котором большая часть частиц сосредоточена в области с максимальными весами w_i .

Описанный метод получения весов частиц достаточно прост и в то же время, как показала практика, позволяет получить хорошие результаты. В качестве иллюстрации на рис. 4 показано полученное в одном из циклов работы алгоритма навигации реальное распределение весов, полученных путём сравнения яркостной картины плоскостей из базы данных и текущего кадра. На этом шаге в дальнометрическом кадре были получены три плоскости, две из которых параллельны. В результате после решения системы уравнений частицы распределились вдоль линии неопределённости, параллельной обеим плоскостям. На этапе вычисления весов по интенсивностям отражённого сигнала были получены значения весов, в распределении которых имеется хорошо выраженный максимум (рис. 4). В результате, после выборки по значимости, частицы сосредоточились возле положения, соответствующего максимуму весов и таким образом неопределённость была разрешена.

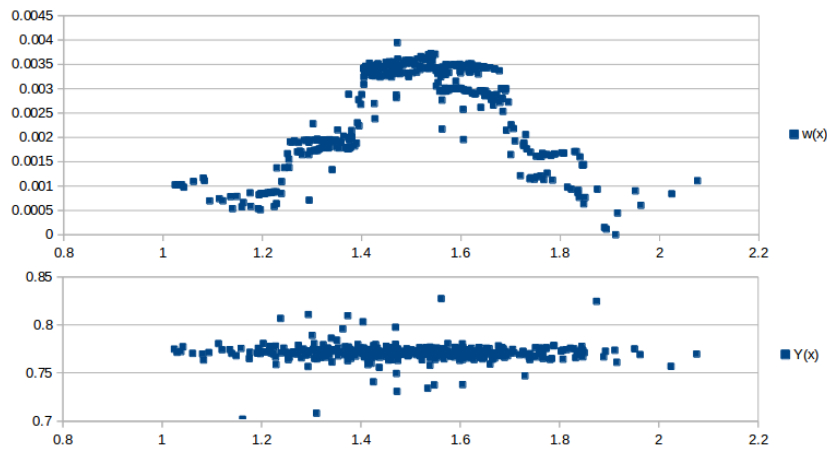


Рис. 4. Определение весов частиц

Формирование дальнометрического кадра. Так как аппарат непрерывно движется и вращается, а дальнометр формирует 2D сканы, то возникает необходимость их объединения в одном 3D дальнометрическом кадре с учётом относительного смещения.

Строки с дальномера поступают с равномерными временными интервалами и сохраняются в буфере, который, после набора заданного числа N сканов (которое зависит от скорости вращения ЛА вокруг вертикальной оси и желаемого сектора обзора), передаётся на дальнейшую обработку. Буфер содержит множество сканов $\{{}^bC_1, {}^bC_2, \dots, {}^bC_N\}$, заданных в СК body, причём строка N – самая поздняя. Одновременно ведётся запись истории преобразований oT и для каждого скана имеется соответствующее его моменту времени преобразование: $\{{}^oT_{C_1}, {}^oT_{C_2}, \dots, {}^oT_{C_N}\}$.

Для формирования дальнометрического кадра необходимо перевести сканы ${}^bC_1, \dots, {}^bC_{N-1}$ в то положение СК body, которое соответствовало времени получения скана bC_N (рис. 5), с помощью преобразования:

$${}^{C_N}T_{C_j} = \left({}^oT_{C_N} \right)^{-1} {}^oT_{C_j}.$$

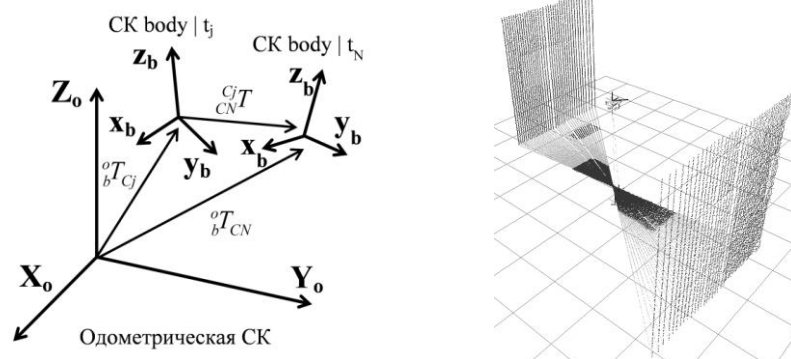


Рис. 5. Формирование дальнометрического кадра

Выделение плоскостей. Для выделения плоскостей используется алгоритм типа RANSAC (RANDOM SAMPLE CONSENSUS) [18] (реализованный в пакете PCL [19]), с помощью которого в исходном облаке точек итерационными методами ищутся подмножества точек, принадлежащие отдельным плоскостям с заданным допуском. Алгоритмы типа RANSAC устойчивы к шумам и могут работать с неструктурированными входными данными, но становятся медлительными при увеличении числа точек в облаке и количества плоскостей в кадре. Существует ряд других способов [2], которые, благодаря структурированности входных данных (что характерно практически для всех существующих сенсоров), позволяют повысить скорость процесса выделения плоскостей.

В результате на этом этапе из исходного облака точек bOT формируется список уравнений плоскостей и соответствующих им точек:

$$\left\{ {}^b\Pi_1 = \{ {}^b\bar{n}_1, {}^b p_1, {}^bOT_1 \}, {}^b\Pi_2 = \{ {}^b\bar{n}_2, {}^b p_2, {}^bOT_2 \}, \dots \right\}.$$

Представление плоскости. После выделения i -ой плоскости (её параметров ${}^b\bar{n}_i$ и ${}^b p_i$) и принадлежащих ей точек OT_i осуществляется процедура описания плоскости, в результате чего с ней связывается СК (определяемая преобразованием ${}^bT_{\Pi_i}$) и формируется множество частей A_i , описывающих дискретные участки плоскости, в которые попали дальнометрические измерения. Таким образом для дальнометрического кадра формируется список плоскостей $\{ {}^b\Pi_1, {}^b\Pi_2, \dots \}$, каждая из которых представлена как ${}^b\Pi_i = \{ {}^b\bar{n}_i, {}^b p_i, {}^bT_{\Pi_i}, A_i \}$.

Формирование множества A_i осуществляется следующим способом (рис. 6). Облако точек ${}^bO T_i$ переводится в СК плоскости Π_i : ${}^{i_i}\bar{P}_j = \left({}^{bT} \right)^{-1} {}^b\bar{P}_j$, и проецируется на плоскость OXY . Плоскость OXY разбита на ячейки заданного размера. Если в какую-либо ячейку попадает точка ${}^{i_i}\bar{P}_j$, то такая ячейка становится частью плоскости и с ней связывается число попавших в неё точек N_T и сумма интенсивностей $\sum I$. Благодаря такому подходу всегда можно получить усреднённое по всем точкам значение интенсивности: $\bar{I}_k = \sum I / N_T$, которое в дальнейшем используется для определения весов w_i частиц.

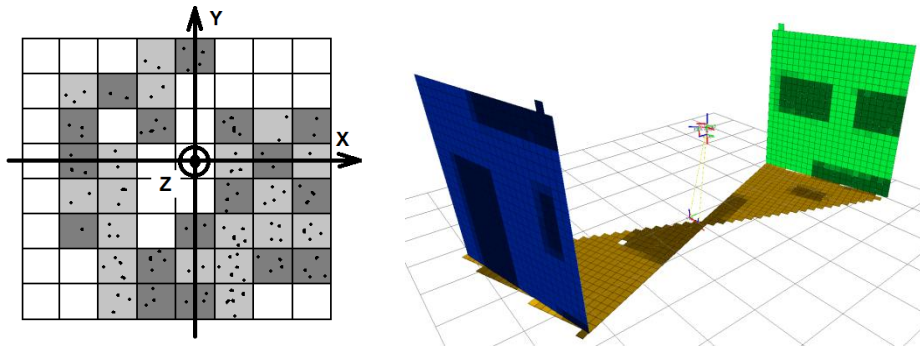


Рис 6. Работа с плоскостями

Обновление БД. После определения положения робота относительно карты, выделенные в новом кадре плоскости, по теперь уже известному преобразованию, вносятся в базу данных. Делается это следующим образом. Для каждой из новых плоскостей, используя её параметрическое уравнение в качестве запроса, в базе данных ищутся все плоскости с близкими (с некоторым допуском) значениями параметров. В общем случае может быть обнаружено более одной плоскости. Найденные плоскости разбиваются на группы по признаку пересечения границ, причём учитывается и то, что плоскости могут пересекаться через другую плоскость-посредник. В каждой из найденных групп плоскости объединяются в одну, и в результирующую плоскость переносится информация из остальных плоскостей с целью её усреднения.

Облака точек, не попавших ни на одну из плоскостей также необходимо сохранить, так как они понадобятся при определении геометрической проходимости пространства при планировании траектории движения и для более детального представления визуальной модели для оператора. Хранить можно непосредственно облака точек, но более рационально использование структур типа восьмеричное дерево, которые используют принцип иерархичного разбиения пространства для более эффективного хранения неорганизованных массивов данных. Этот вопрос подробно рассмотрен в работе [11] (там же представлена ссылка на программную реализацию).

Экспериментальная апробация. Для проверки работоспособности предлагаемого алгоритма была использована подсистема ROS [20] и физический симулятор GAZEBO [21]. Для симулятора, на основе [22], была реализована математическая модель квадрокоптера. Для обработки сигналов инерциальных датчиков и стабилизации полёта были использованы разработанные ранее оригинальные алгоритмы.

Для имитации рабочего пространства была разработана модель помещения (рис. 7), стены которого имеют контрастные элементы, возвращающие разные интенсивности отражённого сигнала дальномера. Общая площадь помещения равна примерно 200 м^2 .

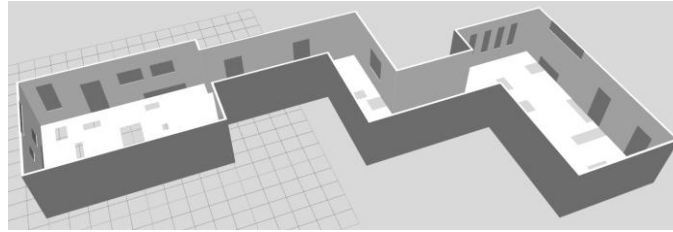


Рис. 7. Модель рабочего пространства

В поставленном эксперименте квадрокоптер, обрабатывая программную траекторию, двигался через всё помещение. В это время, с помощью описанного выше алгоритма SLAM, определялось его положение и формировалась карта окружающего пространства. База данных плоскостей выводилась в наглядном виде в 3D визуализаторе rviz (из пакета ROS).

В процессе полёта и построения карты велась регистрация положений аппарата относительно карты, определённых с помощью разработанного алгоритма SLAM, и их истинных значений, полученных из физического симулятора GAZEBO. Эти данные в последующем были использованы для анализа возникающих при навигации ошибок (рис. 8).

В качестве лазерного дальномера использована модель Hokuyo UTM-30LX со скоростью сканирования 40 строк в секунду и с разрешающей способностью 0.25 градусов при общем угле 270 градусов (1081 точка в одном скане). Каждая точка, помимо измеренной дальности, содержит значение интенсивности отражённого сигнала.

Для формирования кадра в течение примерно одной секунды собираются строки с дальномера. При скорости вращения аппарата 45 градусов в секунду получаются два сектора по 45 градусов, развернутые на 180 градусов по отношению друг к другу (рис. 5). Таким образом, один кадр содержит свыше 40000 точек. Полученное облако точек обрабатывается с помощью алгоритма RANSAC (из библиотеки PCL [19]) с целью выделения плоскостей.

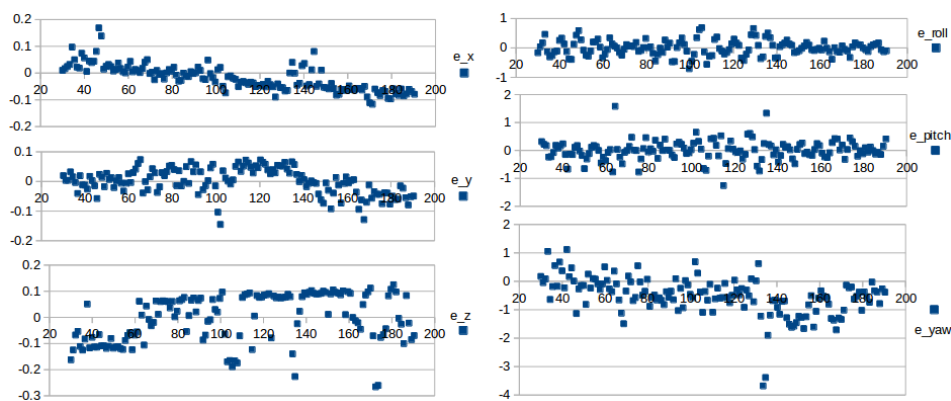


Рис. 8. Значения ошибок при навигации (для x , y , z – метры; для $roll$, $pitch$, yaw – градусы; по горизонтали – время в секундах)

В этом эксперименте было задействовано 400 частиц для аппроксимации плотности распределения вероятности.

Работа выполнялась на платформе (мини-ПК) Intel NUC (модель DC53427) на базе процессора Core i5-3427U (1.8 ГГц), с установленными 128 ГБ SSD накопителем и 16 ГБ ОЗУ. Следует отметить, что специальной оптимизации программы не выполнялось, все основные вычисления алгоритма выполняются в одном потоке.

Для анализа производительности велась запись времени вычислений на каждом из этапов работы алгоритма; их усреднённые по всему маршруту значения представлены в таблице.

Таблица

Усреднённые значения времени выполнения

№	Этап	Среднее время, с
1	Выделение плоскостей	0.061
2	Описание плоскостей	0.020
3	Модель движения для частиц (для всех в сумме)	0.003
4	Решение системы уравнений (для всех в сумме)	0.037
5	Вычисление весов частиц (для всех в сумме)	0.050
6	Определение положения по набору частиц	0.011
Итого		0.182

В результате выполненного в рамках эксперимента полёта была получена модель помещения в виде составляющих её плоскостей (рис. 9).

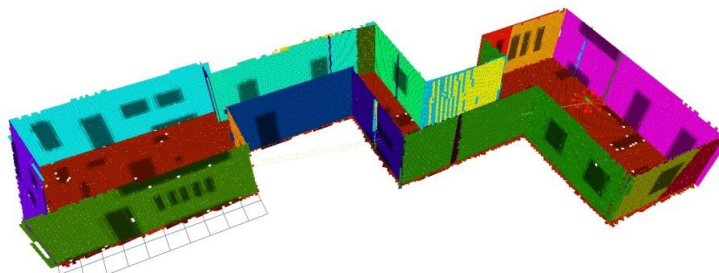


Рис. 9. Построенная карта пространства

Заключение. Полученные в ходе эксперимента результаты показывают работоспособность и эффективность предлагаемого метода. Представление карты в виде геометрических примитивов и усреднение относящейся к одной области пространства информации позволяет существенно сократить объём хранимых в карте данных, повысить скорость работы и стабильность результата. Благодаря использованию яркостной информации данный алгоритм способен разрешать геометрически неопределённые ситуации, а применение вероятностного аппарата повышает его робастность.

Предлагаемый подход обеспечивает переход от больших объёмов исходной информации к компактным описаниям внешней среды, содержащих навигационные данные, что позволяет эффективно решать задачи автономного управления движением мобильных роботов и беспилотных летательных аппаратов с использованием общедоступных вычислительных средств в реальном масштабе времени.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Носков В.П., Носков А.В. Навигация мобильных роботов по дальнометрическим изображениям // Мехатроника, автоматизация, управление. – 2005. – № 12. – С. 16-21.

2. Казьмин В.Н., Носков В.П. Выделение геометрических и семантических объектов в дальнометрических изображениях для навигации роботов и реконструкции внешней среды // Известия ЮФУ. Технические науки. – 2015. – № 10 (171). – С. 71-83.
3. Besl P., McKay N. A method for registration of 3-D shapes // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1992. – № 14. – P. 239-256.
4. Shen S., Michael N., Kumar V. Autonomous Multi-Floor Indoor Navigation with a Computationally Constrained MAV // IEEE International Conference on Robotics and Automation, 2011.
5. Nuchter A., Lingemann K., Hertzberg J., Surmann H. 6 D SLAM–3D Mapping Outdoor Environments // Journal of Field Robotics. – 2007. – No. 24 (8/9). – P. 699-722.
6. Khoshelham K. Automated localization of a laser scanner in indoor environments using planar objects // International Conference on Indoor Positioning and Indoor Navigation. – 2010.
7. Brenner C., Dold C. Automatic relative orientation of terrestrial laser scans using planar structures and angle constraints // ISPRS Workshop on Laser Scanning. – 2007. – P. 84-89.
8. Morris W., Dryanovski I., Xiao J. 3D indoor mapping for micro-UAVs using hybrid range finders and multi-volume occupancy grids // RSS workshop on RGB-D: Advanced Reasoning with Depth Camera, 2010.
9. Schadler M., Stuckler J., Behnke S. Rough terrain 3D mapping and navigation using a continuously rotating 2D laser scanner // German Journal on Artificial Intelligence. – 2014. – No. 28 (2). – P. 93-99.
10. Stuckler J., Behnke S. Multi-resolution surfel maps for efficient dense 3D modeling and tracking // Journal of Visual Communication and Image Representation. – 2014. – No. 25 (1). – P. 137-147.
11. Hornung A., Wurm K., Bennewitz M., Stachniss C., Burgard W. OctoMap: An efficient probabilistic 3D mapping framework based on octrees // Autonomous Robots. – 2013. – Vol. 34, No. 3. – P. 189-206.
12. Hornung A., Osswald S., Maier D., Bennewitz M. Monte Carlo localization for humanoid robot navigation in complex indoor environments // International Journal of Humanoid Robotics. – 2014.
13. Thrun S., Burgard W., Fox D. Probabilistic robotics: Intelligent robotics and autonomous agents. – The MIT Press, 2005. – 672 p.
14. Казьмин В.Н., Носков В. П. Объемное зрение в системе навигационного обеспечения беспилотного летательного аппарата // Вестник МГТУ им. Н.Э. Баумана. Серия «Машиностроение». – 2012. – С. 113-121.
15. Загоруйко С.Н., Казьмин В.Н., Носков В.П. Навигация БПЛА и 3D-реконструкция внешней среды по данным бортовой СТЗ // Мехатроника, автоматизация, управление. – 2014. – № 8. – С. 62-68.
16. Craig John J. Introduction to robotics: mechanics and control. – 3rd ed. – New Jersey: Pearson Education Inc, 2004. – P. 19-54.
17. Sheraz Khan S., Wollherr D., Buss M. Modeling laser intensities for simultaneous localization and mapping // IEEE Robotics and Automation Letters. – 2016. – Vol. 1, No. 2. – P. 692-699.
18. Fischler M., Bolles R. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography // Communications of the ACM. – 1981. – No. 24 (6). P. 381-395.
19. <http://www.pointclouds.org> (дата обращения 29.01.2017).
20. <http://www.ros.org> (дата обращения 29.01.2017).
21. <http://www.gazebosim.org> (дата обращения 29.01.2017).
22. Meyer J., Sendobry A., Kohlbrecher S., Klingauf U., Stryk O. Comprehensive Simulation of Quadrotor UAVs using ROS and Gazebo // Proceedings of the Third international conference on Simulation, Modeling, and Programming for Autonomous Robots. – 2012. – P. 400-411.

REFERENCES

1. Noskov V.P., Noskov A.V. Navigatsiya mobil'nykh robotov po dal'nometricheskim izobrazheniyam [Navigation of mobile robots in telematic images], *Mekhatronika, avtomatizatsiya, upravlenie* [mechatronics, automation, control], 2005, No. 12, pp. 16-21.
2. Kaz'min V.N., Noskov V.P. Vydelenie geometricheskikh i semanticheskikh ob'ektov v dal'nometricheskikh izobrazheniyakh dlya navigatsii robotov i rekonstruktsii vneshney sredy [Detecting geometric and semantic objects in range image for robot navigation and environment reconstruction], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2015, No. 10 (171), pp. 71-83.

3. Besl P., McKay N. A method for registration of 3-D shapes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992, No. 14, pp. 239-256.
4. Shen S., Michael N., Kumar V. Autonomous Multi-Floor Indoor Navigation with a Computationally Constrained MAV, *IEEE International Conference on Robotics and Automation*, 2011.
5. Nuchter A., Lingemann K., Hertzberg J., Surmann H. 6 D SLAM–3D Mapping Outdoor Environments, *Journal of Field Robotics*, 2007, No. 24 (8/9), pp. 699-722.
6. Khoshelham K. Automated localization of a laser scanner in indoor environments using planar objects, *International Conference on Indoor Positioning and Indoor Navigation*, 2010.
7. Brenner C., Dold C. Automatic relative orientation of terrestrial laser scans using planar structures and angle constraints, *ISPRS Workshop on Laser Scanning*, 2007, pp. 84-89.
8. Morris W., Dryanovski I., Xiao J. 3D indoor mapping for micro-UAVs using hybrid range finders and multi-volume occupancy grids // RSS workshop on RGB-D: Advanced Reasoning with Depth Camera, 2010.
9. Schadler M., Stuckler J., Behnke S. Rough terrain 3D mapping and navigation using a continuously rotating 2D laser scanner, *German Journal on Artificial Intelligence*, 2014, No. 28 (2), pp. 93-99.
10. Stuckler J., Behnke S. Multi-resolution surfel maps for efficient dense 3D modeling and tracking, *Journal of Visual Communication and Image Representation*, 2014, No. 25 (1), pp. 137-147.
11. Hornung A., Wurm K., Bennewitz M., Stachniss C., Burgard W. OctoMap: An efficient probabilistic 3D mapping framework based on octrees, *Autonomous Robots*, 2013, Vol. 34, No. 3, pp. 189-206.
12. Hornung A., Osswald S., Maier D., Bennewitz M. Monte Carlo localization for humanoid robot navigation in complex indoor environments, *International Journal of Humanoid Robotics*, 2014.
13. Thrun S., Burgard W., Fox D. Probabilistic robotics: Intelligent robotics and autonomous agents. The MIT Press, 2005, 672 p.
14. Kaz'min V.N., Noskov V.P. Ob'emnoe zrenie v sisteme navigatsionnogo obespecheniya bespilotnogo letatel'nogo apparata [Surround vision in the navigation support system unmanned aerial vehicle], *Vestnik MGTU im. N.E. Baumana. Mashinostroenie* [Herald of the Bauman Moscow State Technical University. Series Mechanical Engineering], 2012, pp. 113-121.
15. Zagoruyko S.N., Kaz'min V.N., Noskov V.P. Navigatsiya BPLA i 3D-rekonstruktsiya vneshney sredy po dannym bortovoy STZ [UAV navigation and 3D reconstruction of environments according to the onboard STZ.] *Mekhatronika, avtomatizatsiya, upravlenie* [Mechatronics, automation, control], 2014, No. 8, pp. 62-68.
16. Craig John J. Introduction to robotics: mechanics and control. 3rd ed. New Jersey: Pearson Education Inc, 2004, pp. 19-54.
17. Sheraz Khan S., Wollherr D., Buss M. Modeling laser intensities for simultaneous localization and mapping, *IEEE Robotics and Automation Letters*, 2016, Vol. 1, No. 2, pp. 692-699.
18. Fischler M., Bolles R. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM*, 1981, No. 24 (6), pp. 381–395.
19. Available at:<http://www.pointclouds.org> (accessed 29 January 2017).
20. Available at:<http://www.ros.org> (accessed 29 January 2017).
21. Available at:<http://www.gazebo.org> (accessed 29 January 2017).
22. Meyer J., Sendobry A., Kohlbrecher S., Klingauf U., Stryk O. Comprehensive Simulation of Quadrotor UAVs using ROS and Gazebo, *Proceedings of the Third international conference on Simulation, Modeling, and Programming for Autonomous Robots*, 2012, pp. 400-411.

Статью рекомендовал к опубликованию д.т.н. С.Г. Цариченко.

Казьмин Вячеслав Николаевич – Московский государственный технический университет им. Н.Э. Баумана; e-mail: slvk@list.ru; 105005, г. Москва, ул. 2-я Бауманская, 5; тел.: +79099050702; кафедра специальной робототехники и мехатроники, аспирант; НИИ Специального машиностроения, инженер.

Kazmin Vyacheslav Nikolaevich – Bauman Moscow State Technical University; e-mail: slvk@list.ru; 5, 2nd Baumanskaya street, Moscow, 105005, Russia; phone: +7 +79099050702; the department of special robotics and mechatronics, post-graduate student; NIISM, engineer.