

Е. А. Девятериков, Б. Б. Михайлов

ВИЗУАЛЬНЫЙ ОДОМЕТР

Описан способ использования системы технического зрения для измерения пройденного пути и углов поворота мобильного робота, основанный на измерении смещения особых точек пространства, информация о которых получается из анализа последовательности изображений.

E-mail: retegor@mail.ru; robot@bmstu.ru.

Ключевые слова: стереозрение, телекамера, система координат, изображение рабочей зоны, особые точки на изображении, ошибки измерения, измерение пути.

Для движения по траектории мобильному роботу необходима система определения его местоположения и ориентации в пространстве. В настоящее время для навигации широко используются энкодеры, инерциальные системы, GPS и дальномеры. Однако использование перечисленных устройств имеет свои недостатки. Навигация с помощью энкодеров (колесная одометрия) проблематична при движении по пересеченной местности, когда возникает проскальзывание колес. Информация, получаемая от инерциальных измерительных устройств, постоянно меняется и нуждается в периодической коррекции. GPS-навигация неточна, требует установки дополнительного оборудования, а также неприменима для работы в экстремальных условиях (например, в подвале). Система на основе дальномеров очень сложна. Она требует предварительного знания окружающей обстановки, чтобы сравнивать расстояния до одного и того же объекта в разные моменты времени. Кроме того, дальномер имеет высокую стоимость и нуждается в сложной системе обработки информации.

Метод визуальной одометрии основан на измерении смещения особых (ключевых) точек пространства, координаты которых находятся из анализа последовательности стереоизображений. Такая система нечувствительна к проскальзыванию колес и может работать в недетерминированной, т. е. в заранее не подготовленной среде, пассивна и универсальна как человеческое зрение. К тому же наличие системы технического зрения позволяет параллельно решать множество сопряженных задач, таких как локализация и построение карты пространства.

Алгоритм визуальной одометрии. Изображения одного и того же объекта, наблюдаемого двумя камерами, расположенными на определенном расстоянии друг от друга, отличаются. Сопоставляя эти два

изображения можно находить трехмерные координаты точек пространства. Идея алгоритма визуальной одометрии [1–3] заключается в том, чтобы выделять на стереоизображениях набор точек, обладающих определенным свойством, например углы зданий, темные пятна на асфальте и т. д., наблюдать изменение их положения, а затем пересчитать это изменение в смещение и угол поворота робота [4].

Работа алгоритма выполняется в следующей последовательности:

- 1) ввод 1-й стереопары, удаление искажений, выравнивание эпиполярных линий;
- 2) поиск особых точек на левом изображении и этих же точек на правом;
- 3) вычисление пространственных координат этих точек;
- 4) ввод 2-й стереопары, удаление искажений, выравнивание эпиполярных линий;
- 5) поиск особых точек из предыдущего левого изображения на текущем;
- 6) поиск особых точек текущего правого изображения на предыдущем;
- 7) вычисление трехмерных координат точек текущей стереопары;
- 8) сравнение координат точек в текущий и предыдущий моменты времени и вычисление матриц поворота и переноса камеры;
- 9) переход к п. 4.

Рассмотрим основные операции алгоритма.

Выравнивание изображений. Все вводимые стереоизображения проходят предварительную обработку, основным элементом которой является выравнивание. В результате каждая точка левого изображения и соответствующая ей точка правого изображения должны лежать на одной горизонтальной прямой. Выравнивание стереоизображений включает три этапа [5]:

- удаление радиальных искажений;
- совмещение эпиполярных линий двух изображений [6];
- выделение информационной части изображений.

Выравнивание позволяет находить точки на правом изображении, точно соответствующие точкам левого изображения.

Поиск особых точек. Особая точка сцены (точечная особенность) – это такая точка, изображение которой можно устойчиво отличать от изображений всех соседних с ней точек. Чтобы вычислить смещение робота, необходимо выделить и определить координаты особых точек сцены.

На сегодняшний день существует множество методов выделения особых точек. Подавляющее большинство таких алгоритмов анализирует функцию интенсивности в окрестности некоторой точки и сравнивает ее значение с пороговым.

В методе Гарриса (Harris) [7] для окрестности каждой точки находится матрица вторых производных от интенсивности яркости изображения (I) – матрица Гессе:

$$M(I) = \begin{bmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial x \partial y} \\ \frac{\partial^2 I}{\partial x \partial y} & \frac{\partial^2 I}{\partial y^2} \end{bmatrix}.$$

Согласно методу Гарриса, точки располагаются там, где автокорреляционная матрица имеет два больших собственных значения, например: если в двух направлениях от точки резко меняется интенсивность яркости, то найден угол. Далее вычисляется функция-критерий R и ее значение сравнивается с пороговым:

$$R = \det(M) - k(\text{trace}(M))^2, \quad k = 0,04 \dots 0,06.$$

Отметим, что вторые производные используются для того, чтобы находить особые точки, инвариантные к освещению и повороту изображения, что полезно при их отслеживании.

В методе ускоренной проверки сегментов (Features from Accelerated Segment Test – FAST) яркости точек, образующих окружность заданного радиуса, сравниваются с яркостью точки, лежащей в центре этой окружности [8] (рис. 1).

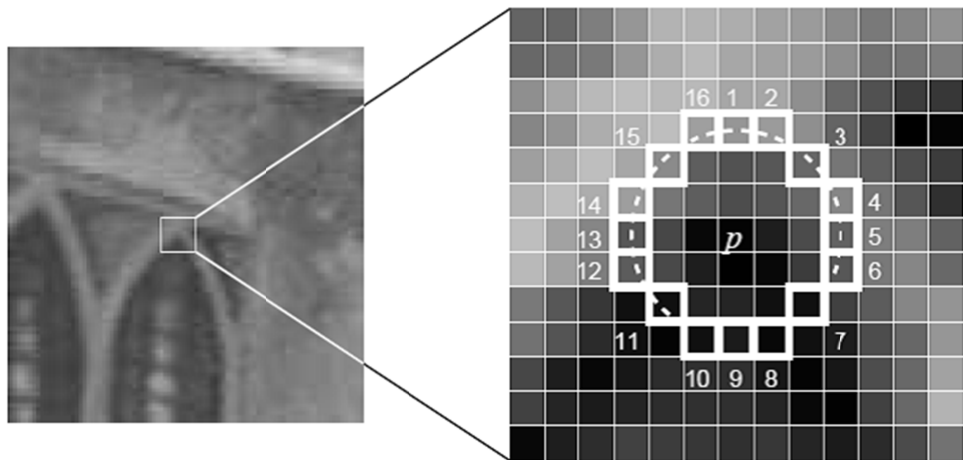


Рис. 1. Иллюстрация работы алгоритма FAST

Как правило, радиус окружности составляет 3 пикселя, т. е. оцениваются всего 16 точек. Если яркость по крайней мере N последовательных точек рассматриваемой окружности больше или меньше яркости ее центра на заданное пороговое значение, то центр окруж-

ности считается особой точкой. В результате многочисленных тестов авторами метода было установлено, что самым эффективным является модификация FAST-9, в которой $N = 9$.

Алгоритм FAST позволяет находить точки, инвариантные к повороту. Это один из самых быстрых методов выделения особых точек на сегодняшний день, который позволяет работать в масштабе реального времени. Время обработки изображения алгоритмом FAST в 5 раз меньше, чем у ближайшего аналога SUSAN, и в 20 раз меньше, чем у традиционно используемого в техническом зрении алгоритма Гарриса.

Ускоренный метод нахождения особых точек (Speeded-Up Robust Features – SURF) [9] относится к лидерам по надежности. С его помощью находят особые точки изображения, инвариантные к масштабированию, повороту, искажениям и освещению. Метод включает нахождение дескриптора особенностей – обобщенной характеристики окрестности точки, позволяющей сопоставлять особенности двух изображений.

В методе SURF используется масштабируемое гауссово пространство изображений $f : R^2 \rightarrow R$, семейство $\{L(x, y, t) : R^2 \cdot R_+ \rightarrow R\}$ постепенно сглаживающихся версий исходного изображения

$$L(x, y, t) = g(x, y, t) * f(x, y),$$

где $*$ – операция свертки; $g(x, y, t)$ – двумерная функция Гаусса со стандартным отклонением $t = \sigma^2$, называемым масштабом. В результате выделяются особые точки, инвариантные к масштабированию изображения.

Детектор особых точек основан на матрице Гессе:

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix},$$

где $L_{xx}(x, \sigma) = \frac{\partial g(x, y, t)}{\partial x \partial x} * I(x, y)$; $L_{xy}(x, \sigma) = \frac{\partial g(x, y, t)}{\partial x \partial y} * I(x, y), \dots$

Критерием выделения особенностей является определитель матрицы Гессе

$$\det(H) = L_{xx}L_{yy} - (\omega L_{xy})^2,$$

где $L_{ij}(L_{xx}, L_{xy}, L_{yy})$ – свертки для текущей точки в текущем масштабе; ω – коэффициент, компенсирующий сжатие при аппроксимации (для сжатия в 1,2 раза $\omega = 0,9$).

Дескриптор состоит из 64 значений. Окрестность особой точки разбивается на 16 квадрантов. Для каждого квадранта находят четыре значения: суммы производных интенсивности в двух направлениях и суммы модулей производных интенсивности в двух направлениях.

Затем простым перебором, методом RANSAC или построением $k-d$ -дерева сравниваются значения дескрипторов особых точек двух изображений.

Слежение за особыми точками методом Люкаса и Кенедэ. Для вычисления пути робота находят смещение особых точек между текущим и предыдущим изображением. Алгоритм [10] имеет следующие ограничения: постоянство яркости и когерентное смещение всех точек. При этом само смещение должно быть небольшим. Условия выполняются при достаточно высокой частоте обработки изображений.

Пусть x – особенность первой функции F , найдем такую точку $x+h$ функции G , в которой расстояние между окрестностями этих точек минимально.

Расстояние между окрестностями записывается в виде

$$E = \sum_{x \in R} [F(x+h) - G(x)]^2,$$

где в нашем случае $F(x)$, $G(x)$ – интенсивность точки с координатами (x, y) на первом и втором изображениях. Функцию $F(x)$ раскладывают в ряд Тейлора:

$$F(x+h) = F(x) + h \frac{\partial}{\partial x} F(x),$$

где $\frac{\partial}{\partial x}$ – градиент (для изображений по площади). Далее ищем минимум E :

$$\begin{aligned} 0 &= \frac{\partial}{\partial h} E \approx \frac{\partial}{\partial h} \sum_x \left[F(x) + h \frac{\partial F}{\partial x} - G(x) \right]^2 = \\ &= \sum_x 2 \frac{\partial F}{\partial x} \left[F(x) + h \frac{\partial F}{\partial x} - G(x) \right]. \end{aligned}$$

Отсюда смещение h можно получить как

$$h = \left[\sum_x \left(\frac{\partial F}{\partial x} \right)^T [G(x) - F(x)] \right] \left[\sum_x \left(\frac{\partial F}{\partial x} \right)^T \left(\frac{\partial F}{\partial x} \right) \right]^{-1}.$$

Определение расстояния до особых точек. Рассмотрим самый простой случай, когда камеры расположены параллельно друг другу. Фокусные расстояния f известны и вычисляются при калибровке камер [8]. Расстояние Z до точки P обратно пропорционально разнице между координатами проекций этой точки на левом и правом изображениях $d = x^l - x^r$:

$$Z = \frac{fT}{x^l - x^r}, \quad (1)$$

где T – расстояние между камерами (база); f – фокусное расстояние; x^l и x^r – координаты проекций точки на левом и правом изображениях.

Величина d называется рассогласованием (disparity). Так как расстояние до точек обратно пропорционально d , то очевидно, что зависимость нелинейна. Когда d близко к 0, значительно увеличивается ошибка вычислений, поэтому до геометрических расчетов допускаются точки с относительно большим d (>5 % ширины изображения, т. е. близкие точки).

Формула (1), по которой вычисляется расстояние до точки пространства, верна только в том случае, если координаты x^l и x^r проекций точки получены в один и тот же момент времени. В действительности же это не так, особенно если в системе отсутствует устройство синхронизации работы телекамер и устройств видеозахвата. На рис. 2 показана схема для оценки ошибки измерения расстояния до точек за счет рассогласования времени захвата кадров с двух телекамер.

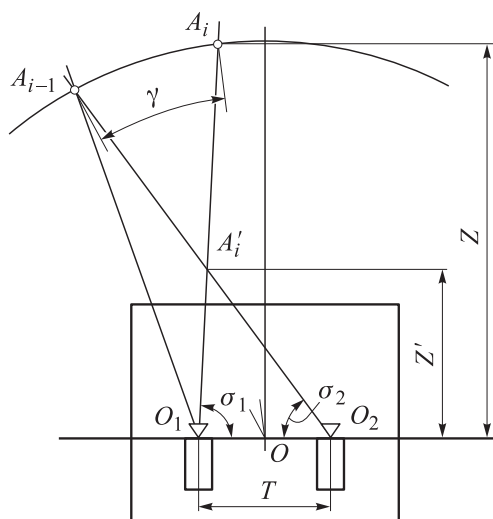


Рис. 2. Геометрическая схема для оценки ошибки вычисления расстояния

Здесь O_1 и O_2 – оптические центры левой (первой) и правой (второй) телекамер. Положение точки пространства A_{i-1} определяется как пересечение лучей O_1A_{i-1} и O_2A_{i-1} , направление которых (углы σ_1 и σ_2) известно из положения проекций точки на изображениях. Пусть за некоторый интервал времени $t_i \dots t_{i-1}$ робот повернулся на угол γ . Пусть также в i -й момент времени, когда был сохранен кадр с первой телекамеры, точка A в системе координат робота находилась в положении A_i . Вторая телекамера зафиксировала точку A в $(i-1)$ -й момент времени в другом положении A_{i-1} . Тогда ее положение ошибочно будет определено как A'_i .

Результаты геометрического расчета показывают, что при повороте робота с угловой скоростью $\omega = \pi/2$ рад/с и рассогласовании времени захвата изображений 0,033 с (отставание на один кадр при частоте видеозахвата 30 с^{-1}) расстояние до точки пространства вычисляется с ошибкой порядка 35 %. Ошибка 5 % возникает при повороте робота с угловой скоростью $\omega = \pi/20$ рад/с.

Интерлейсинг и смазывание изображений также негативно влияют на работу визуального одометра. Когда робот перемещается вперед, вдоль оси камер, изображения не искажаются. Но при повороте робота появляется эффект «гребенки» (интерлейсинг): четные строки изображения отстают от нечетных, что является следствием использования в видеосистемах чересстрочной развертки. При повороте робота со скоростью $15^\circ/\text{с}$ выделение особых точек становится невозможным. Данную проблему можно решить двумя способами:

- вычислить величину смещения строк и совместить их. Но эта задача по ресурсоемкости соизмерима со всей задачей визуальной одометрии;

- удалить четные строки изображения (чересстрочная фильтрация), что снизит разрешение изображений по вертикали, но скажется на точности нахождения угла тангажа робота.

После применения чересстрочного фильтра метод FAST нормально функционирует при повороте робота со скоростью до $30^\circ/\text{с}$. При повороте робота с большей скоростью даже после фильтрации негативного эффекта интерлейсинга изображение остается «смазанным» из-за недостаточного быстродействия системы технического зрения, а алгоритмы поиска особых точек не функционируют.

Адаптивная бинаризация изображения позволяет избавиться от пикселей средней яркости, т. е. от «смазанных» участков, а затем усилить контуры изображения.

Вычисление перемещения робота. Цель алгоритма визуальной одометрии – определить положение робота в текущий момент време-

ни относительно его положения в момент запуска алгоритма. Все предшествующие вычисления направлены на получение трехмерных координат точек пространства, в котором находится робот, относительно робота. Задача блока вычисления перемещения – определить по этим координатам величину перемещения робота. На рис. 3 показано перемещение системы координат робота, где система координат $O^O X^O Y^O Z^O$ неподвижна; $O^C X^C Y^C Z^C$ и $O^P X^P Y^P Z^P$ – положение системы координат робота в текущий и в некоторый предыдущий моменты; M_i – неподвижные особые точки пространства.

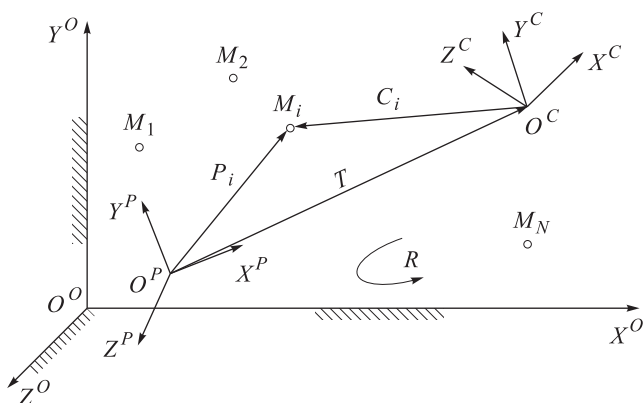


Рис. 3. Перемещение системы координат робота

Пусть координаты i -й особой точки пространства в системе координат робота в текущий момент времени $C_i = \begin{pmatrix} X_i^C & Y_i^C & Z_i^C \end{pmatrix}^T$, координаты той же точки пространства в некоторый предыдущий момент времени $P_i = \begin{pmatrix} X_i^P & Y_i^P & Z_i^P \end{pmatrix}^T$. Пусть также найдено N особых точек. Тогда данные образуют выборку пар векторов $W = ((P_1, C_1), \dots, (P_N, C_N))$ объемом N . Изменение положения системы координат робота будем описывать в виде аффинного преобразования

$$f: R^3 \rightarrow R^3, f(x) = Rx + T,$$

где $T = \begin{pmatrix} \Delta X & \Delta Y & \Delta Z \end{pmatrix}^T$ – вектор переноса; $R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$ –

матрица поворота.

Задача заключается в получении оценки вектора переноса T и матрицы поворота R преобразования системы координат робота на основе выборки W .

Поскольку изменение положения системы координат робота описывается аффинным преобразованием, координаты i -й точки пространства в текущий и предыдущий момент времени связаны выражением

$$C_i = RP_i + T,$$

или

$$\begin{pmatrix} X_i^C \\ Y_i^C \\ Z_i^C \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} X_i^P \\ Y_i^P \\ Z_i^P \end{pmatrix} + \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix}.$$

Выполнив перемножение и сложение матриц, для i -й особой точки получим

$$\begin{pmatrix} X_i^C \\ Y_i^C \\ Z_i^C \end{pmatrix} = \begin{pmatrix} r_{11}X_i^P + r_{12}Y_i^P + r_{13}Z_i^P + \Delta X \\ r_{21}X_i^P + r_{22}Y_i^P + r_{23}Z_i^P + \Delta Y \\ r_{31}X_i^P + r_{32}Y_i^P + r_{33}Z_i^P + \Delta Z \end{pmatrix}.$$

Для нахождения неизвестных решим три системы линейных алгебраических уравнений:

$$X_i^C = r_{11}X_i^P + r_{12}Y_i^P + r_{13}Z_i^P + \Delta X, \quad i = \overline{1, N}; \quad (2)$$

$$Y_i^C = r_{21}X_i^P + r_{22}Y_i^P + r_{23}Z_i^P + \Delta Y, \quad i = \overline{1, N}; \quad (3)$$

$$Z_i^C = r_{31}X_i^P + r_{32}Y_i^P + r_{33}Z_i^P + \Delta Z, \quad i = \overline{1, N}. \quad (4)$$

Системы линейных алгебраических уравнений (2–4) имеют общий вид:

$$x_i^C = r_1 X_i^P + r_2 Y_i^P + r_3 Z_i^P + \Delta t, \quad i = \overline{1, N}. \quad (5)$$

Система (5) содержит четыре неизвестных, поэтому для ее решения необходимо иметь данные по крайней мере о четырех точках. Запишем (5) в матричной форме:

$$A\omega = y, \quad (6)$$

$$\text{где } A = \begin{pmatrix} X_1^P & Y_1^P & Z_1^P & 1 \\ X_2^P & Y_2^P & Z_2^P & 1 \\ \vdots & \vdots & \vdots & \vdots \\ X_N^P & Y_N^P & Z_N^P & 1 \end{pmatrix}; \quad \omega = (r_1, r_2, r_3, \Delta t); \quad y = (x_1^C, x_2^C, \dots, x_N^C)^T.$$

Здесь матрица A не является квадратной, поэтому утверждение $\omega = A^{-1}y$ неверно. Речь идет о задаче регрессионного анализа. Необходимо на основе всей выборки W объемом N получить неизвестные параметры $r_1, r_2, r_3, \Delta t$ регрессионной модели (гипотезы), удовлетворяющей некоторому критерию качества.

В системе уравнений (5) заданы линейная регрессионная модель (функция восстановления значений зависимой переменной)

$$f(r_1, r_2, r_3, \Delta t, X_i^P, Y_i^P, Z_i^P, 1) = r_1 X_i^P + r_2 Y_i^P + r_3 Z_i^P + \Delta t \cdot 1$$

с параметрами ω , а также вектор фактических значений y и выборка значений аргументов A .

Один из способов получения неизвестных параметров r_1, r_2, r_3 и Δt , приближающих значения правых частей всех уравнений системы (5) к значениям левых, – метод наименьших квадратов, который позволяет найти такие оптимальные параметры ω линейной регрессии, чтобы сумма ошибок (регрессионных остатков) была минимальна. Метод заключается в минимизации евклидова расстояния $|A\omega - y|$ между двумя векторами: вектором восстановленных значений зависимой переменной $A\omega$ и вектором фактических значений зависимой переменной y .

Запишем критерий качества модели (функция невязки)

$$S = \sum_{i=1}^N (f(\omega, A_i) - y)^2 = |A\omega - y|^2 \rightarrow \min.$$

Запишем функцию невязки в виде

$$\begin{aligned} S = |A\omega - y|^2 &= (A\omega - y)^T (A\omega - y) = y^T y - y^T A\omega - \\ &- \omega^T A^T y - \omega^T A^T A\omega = y^T y - 2y^T A\omega + \omega^T A^T A\omega. \end{aligned}$$

Чтобы найти минимум функции невязки, найдем ее производную и приравняем к нулю:

$$\begin{aligned} \frac{\partial S}{\partial \omega} &= -2A^T y + 2A^T A\omega = 0; \\ 2A^T A\omega &= 2A^T y; \\ (A^T A)\omega &= (A^T y). \end{aligned} \tag{7}$$

Таким образом, вместо системы уравнений (6) с неквадратной матрицей A будем решать систему уравнений (7) с квадратной матрицей $(A^T A)$, для которой

$$\omega = (A^T A)^{-1} (A^T y).$$

3D-модель для отладки программы. На основе алгоритма визуальной одометрии с использованием приведенных в статье процедур выравнивания и фильтрации изображений, поиска особых точек и вычисления перемещения робота была написана программа. Для ее предварительного тестирования и отладки была создана трехмерная модель в среде SolidWorks. Модель представляет собой сцену (рис. 4), состоящую из двух виртуальных камер, стола и простых объектов (кубиков, «обтянутых» текстурами).

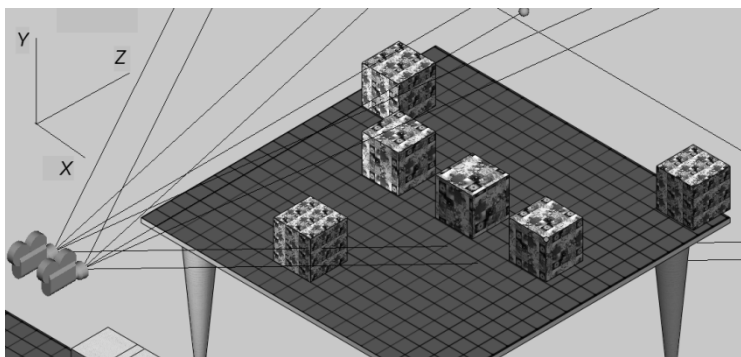


Рис. 4. Модель системы

Сначала программно камеры смещались на равные интервалы по координате X и рассчитывалось среднее квадратичное отклонение вычисленных перемещений, которое при обработке 100 ключевых точек составило 1,8 % всего пути. Затем камеры перемещались по замкнутой траектории, представляющей собой ромб. Погрешность выхода в начальную точку составила 5 % относительно суммарного пройденного пути. Таким образом, была установлена работоспособность алгоритма.

Экспериментальное исследование системы. Для экспериментальной проверки программы, реализующей алгоритм визуальной одометрии, был изготовлен макет робота (рис. 5).

Состав системы:

- видеокамеры Sony 640x480 пикс. (2 шт.);
- компьютер Intel i5 2.4GHz, 2Gb DDR3 (используется 1 поток);
- устройства видеозахвата Pinnacle DVC100 (2 шт.);
- мобильная платформа.

Камеры были закреплены параллельно полу и параллельно друг другу, расстояние между осями объективов (база) – 120 мм.

Вычислялись только две координаты – пройденный путь и угол поворота (курс) робота.

Для оценки точности системы и алгоритмов поиска особых точек несколько раз был обработан один и тот же набор из 52 пар изображений, полученный при смещении робота вперед на равные расстояния (рис. 6). В процессе обработки оценивалось среднее квадратичное отклонение найденной траектории от действительной относительно длины пути макета:

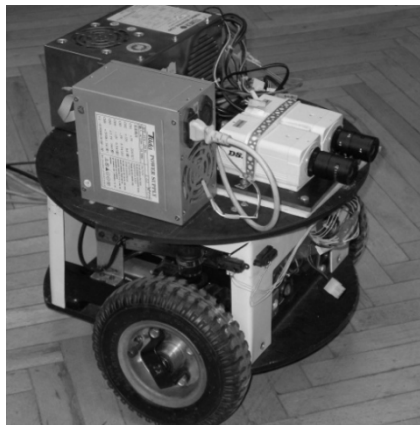


Рис. 5. Система визуальной одометрии на мобильном роботе

$$c = \frac{\sqrt{\frac{\sum_{i=0}^N (Z_i - Z')^2}{N}}}{L} \cdot 100 \%,$$

где Z_i – найденное перемещение за один сдвиг макета; Z' – действительное перемещение за один сдвиг макета; N – число кадров (обработанных стереопар), $N = 52$; L – действительная длина пути.

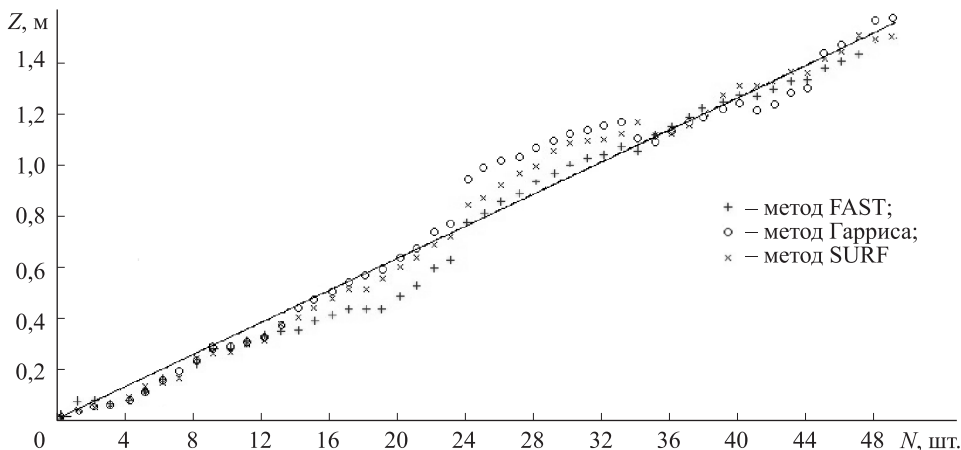


Рис. 6. Результаты измерения при смещении робота на равные интервалы (Z – пройденный путь)

Затем робот несколько раз перемещался по замкнутой траектории: 1,5 м вперед, 1,5 м назад, после чего оценивалась средняя ошибка выхода в исходную точку [11] (рис. 7).

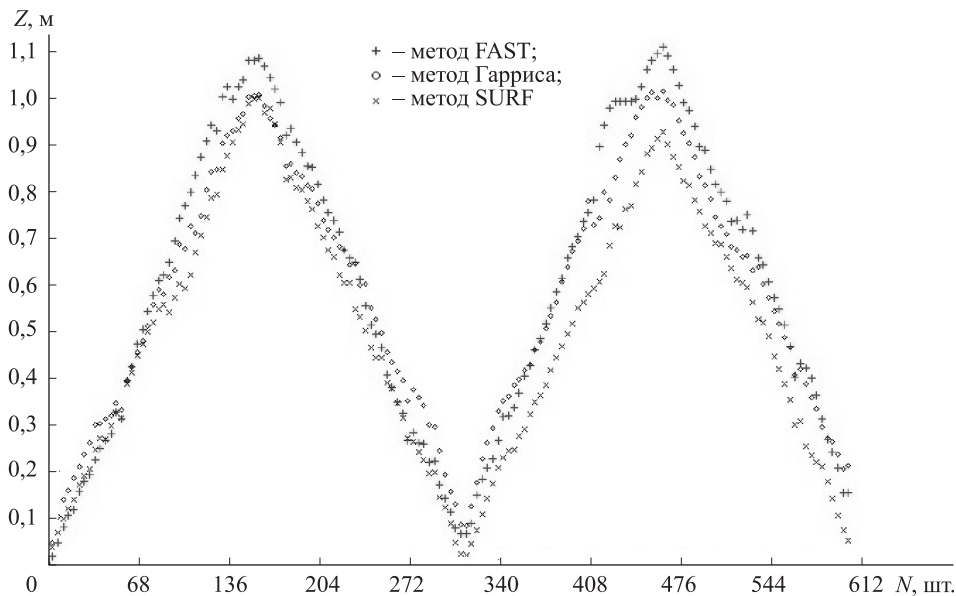


Рис. 7. Результаты измерения при перемещении робота по замкнутой траектории

В табл. 1 приведены скорость обработки и ошибки при использовании трех методов поиска особых точек.

Таблица 1

Сводная таблица результатов эксперимента

| Метод выделения особых точек | Скорость обработки, стереопар/с | Ошибка выхода в исходную точку, % | Среднее квадратичное отклонение при смещении на равные интервалы, % |
|------------------------------|---------------------------------|-----------------------------------|---------------------------------------------------------------------|
| FAST | 37,2 | 7,2 | 4,8 |
| Гарриса | 17,9 | 7,6 | 5,2 |
| SURF | 10,0 | 3,8 | 4,1 |

Затем обрабатывались записи, полученные при повороте робота со скоростью около 25 °/с, анализировалась средняя точность выхода угла в начальное значение для каждой итерации. На рис. 8 представлены графики изменения вычисленного угла с применением различных фильтров, а в табл. 2 – численные значения результатов эксперимента.

Таблица 2

Результаты при нормальной скорости поворота робота

| Фильтр изображения | Скорость обработки (FPS), кадров/с | Ошибка выхода в исходную точку, % |
|--------------------|------------------------------------|-----------------------------------|
| Без фильтров | 37,2 | 20,1 |
| Чересстрочный | 33,0 | 10,8 |
| С бинаризацией | 22,0 | 12,6 |

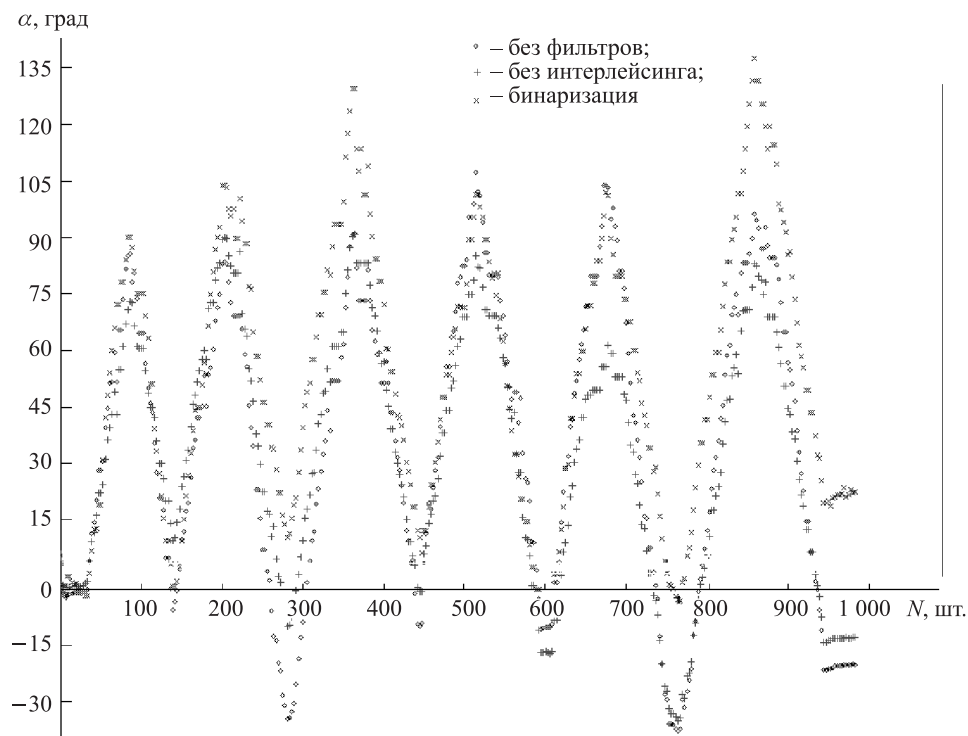


Рис. 8. Результаты измерения угла поворота робота (α – угол поворота робота; N – количество обработанных пар изображений)

Результаты эксперимента показывают, что метод визуальной одометрии позволяет достаточно точно измерить пройденный путь и угол поворота робота. При ошибке 3,8 % самую высокую точность и низкую производительность (10 кадров/с) обеспечивает метод SURF. В режиме реального времени может работать метод FAST, дающий возможность обрабатывать 37 пар кадров за секунду, но дающий ошибку 7,2 %. Система позволяет также измерить угол поворота робота с ошибкой 10 %.

Эксперимент показал, что удаление четных/нечетных строк из изображения незначительно снижает точность измерений при медленном перемещении робота, когда нет негативного эффекта интерлейсинга. Но способ чересстрочной фильтрации принципиально улучшает точность работы системы при повороте робота с рабочей скоростью, поскольку без представленного фильтра ошибка измерения достигает 20 %. Усиление контуров изображения с помощью адаптивной бинаризации снижает точность системы, но дает возможность вычислять угол поворота робота при вращении со скоростью более 35 %/с.

В процессе эксперимента были решены следующие задачи: разработана структура системы визуальной одометрии, реализован алгоритм визуальной одометрии в виде программы, создана трехмерная

модель системы для проверки работоспособности алгоритма, изготовлен макет системы, проверена работоспособность системы на реальных изображениях, полученных при движении мобильного робота, определены ее точность и быстродействие.

В дальнейшей работе будут опробованы различные методы обработки получаемых массивов особых точек для более качественной оценки реального перемещения, например RANSAC. Метод SURF будет оптимизирован под решаемую задачу.

СПИСОК ЛИТЕРАТУРЫ

1. Jin Xu, Minyi Shen, Weiqiang Wang, Li Yang, Jilin Liu, Kalyaev I. Robust stereo visual odometry for autonomous rover. Proceedings of 6th WSEAS International Conference on Signal, Speech and Image Processing, 2006.
2. Nister D., Naroditsky O., Bergen J. Visual odometry for ground vehicle applications // Journal of Field Robotics, 2006. Iss. 23(1). P. 3–20.
3. Beets E., Boukir S., Suter D. Aircraft Pose Estimation from Homography. Technical report MECSE-1-2004, Monash University, 2004.
4. DeMenthon D.F., Davis L.S. Model-based object pose in 25 lines of code. Proceedings of the European Conference on Computer Vision, 1992.
5. Bradski G., Kaehler A. Learning OpenCV. Newgen Publishing and Data Services – 2008.
6. Ко ну шин А. Геометрические свойства нескольких изображений // Компьютерная графика и мультимедиа. – 2006. – Вып. № 4(3). – <http://cgm.computergraphics.ru/content/view/141>.
7. Га га нов В. Инвариантные алгоритмы сопоставления точечных особенностей на изображениях // Компьютерная графика и мультимедиа. – 2009. – Вып. № 7(1). – http://cgm.computergraphics.ru/issues/issue17/invariant_features.
8. Rosten E., Drummond T. Points and Lines for High Performance Tracking. International Conference on Computer Vision. 2005. ISS 2 (Oct.). P. 1508–1511.
9. Bay H., Ess A., Tuytelaars T., Gool V. Speeded-Up Robust Features (SURF). Computer vision and image understanding (CVIU). 2008. Vol. 110(3). P. 346–359.
10. Ко ну шин А. Слежение за точечными особенностями сцены (Point feature tracking), 2003. <http://cgm.computergraphics.ru/content/view/54>.
11. Jang G., Kim S., Lee W., Kweon I. Color Landmark Based Self-Localization for Indoor Mobile Robots. Korea Advanced Institute of Science and Technology, 2001.

Статья поступила в редакцию 28.06.2012