

УДК 629.7.058

<https://doi.org/10.32362/2500-316X-2021-9-2-22-34>

НАУЧНАЯ СТАТЬЯ

Пешеходная навигация: чем инерциальные модули могут помочь смартфонам?

И.А. Чистяков^{1,®}, И.В. Гришов², А.А. Никулин³,
М.В. Пихлецкий⁴, И.Б. Гарцеев^{2,4}

¹ Московский государственный университет, Москва, 119234 Россия

² МИРЭА – Российский технологический университет, Москва, 119454 Россия

³ ООО «КС Кадровый Консалтинг», Москва, 115191 Россия

⁴ ООО «Техкомпания Хуавэй», Москва, 121614 Россия

® Автор для переписки, e-mail: chistyakov.ivan@yahoo.com

Резюме. Работа посвящена построению референсных траекторий ходьбы человека с целью дальнейшей разработки на их основе алгоритмов пешеходной навигации для смартфонов, в том числе с помощью методов машинного обучения. Рассматривается задача восстановления замкнутых траекторий по данным, полученным с помощью инерциальных измерительных блоков (ИИБ), зафиксированных на ногах в области подъема стопы. Особенности подхода являются использование недорогих датчиков и простота представленного метода. Предлагаются алгоритмы, позволяющие построить сглаженную двумерную траекторию движения пешехода как по измерениям одного ИИБ, так и по совместным измерениям двух блоков. Алгоритмы основаны на использовании модификации фильтра Калмана и предположения о нулевой скорости ИИБ в момент соприкосновения стопы пешехода с поверхностью. В случае двух измерительных блоков дополнительно предполагается, что положения датчиков левой и правой ног не могут значительно отличаться друг от друга. Работа алгоритмов была проверена на траекториях длительностью от 1 до 10 минут, полученных при движении пешеходов внутри помещений по ровным горизонтальным поверхностям. Для оценки полученных результатов восстановленные указанными способами траектории сравниваются с высокоточными решениями, построенными с помощью данных от GNSS-приемников, работающих в RTK-режиме. Также рассматривается вопрос синхронизированного сбора данных от всех источников и приводится подробное описание проведенных экспериментов и используемого оборудования. Набор данных, на котором происходила верификация алгоритмов, свободно доступен по адресу: <http://gartsev.ru/projects/rtj2021>.

Ключевые слова: пешеходная навигация, инерциальная навигация, ИИБ, наблюдение нулевой скорости, инерциальная навигационная система, RTK-режим, датасет, машинное обучение, глубокое обучение, смартфон, синхронизация измерительных блоков

• Поступила: 04.09.2020 • Доработана: 13.01.2021 • Принята к опубликованию: 19.01.2021

Для цитирования: Чистяков И.А., Гришов И.В., Никулин А.А., Пихлецкий М.В., Гарцеев И.Б. Пешеходная навигация: чем инерциальные модули могут помочь смартфонам? *Российский технологический журнал*. 2021;9(2):22–34 <https://doi.org/10.32362/2500-316X-2021-9-2-22-34>

Прозрачность финансовой деятельности: Никто из авторов не имеет финансовой заинтересованности в представленных материалах или методах.

Авторы заявляют об отсутствии конфликта интересов.

RESEARCH ARTICLE

Pedestrian navigation: how can inertial measurement units assist smartphones?

Ivan A. Chistyakov^{1,*}, Ivan V. Grishov², Alexey A. Nikulin³,
Mikhail V. Pikhletsy⁴, Ilya B. Gartsev^{2,4}

¹ Moscow State University, Moscow, 119234 Russia

² MIREA – Russian Technological University, Moscow, 119454 Russia

³ KS Kadrovyyi Consulting, Moscow, 115191 Russia

⁴ Huawei, Moscow, 121614 Russia

* Corresponding author, e-mail: chistyakov.ivan@yahoo.com

Abstract. This paper is devoted to construction of reference walking trajectories for developing pedestrian navigation algorithms for smartphones. Such trajectories can be used both for verification of classical algorithms of navigation or for application of machine learning technics. Reconstruction of closed trajectories based on data from foot-mounted inertial measurement units (IMU) is investigated. The advantages of the approach are the use of inexpensive sensors and the simplicity of the presented method. We propose algorithms for reconstruction of smooth 2D pedestrian trajectories based on measurements from a single IMU as well as on combined measurements from two IMU's. Introduced algorithms are based on application of modified Kalman filter with an assumption of IMU having zero velocity when foot contacts the ground. In case of two measurement units, it is additionally assumed that the positions of the sensors cannot differ significantly from each other. The algorithms were tested on trajectories lasting from 1 to 10 minutes, passing indoors on horizontal surfaces. Obtained results were compared with high precision trajectories acquired with GNSS RTK receivers. Additionally, the process of inter-device time synchronization is investigated and detailed description of the experiments and used equipment is given. The dataset used for verification of proposed algorithms is freely available at: <http://gartsev.ru/projects/rtj2021>.

Keywords: pedestrian navigation, inertial navigation, inertial navigation system (INS), inertial measurement unit (IMU), zero velocity update (ZUPT), foot mounted device, RTK, dataset, machine learning, deep learning, PDR, smartphone, synchronization

• Submitted: 04.09.2020 • Revised: 13.01.2021 • Accepted: 19.01.2021

For citation: Chistyakov I.A., Grishov I.V., Nikulin A.A., Pikhletsy M.V., Gartsev I.B. Pedestrian navigation: how can inertial measurement units assist smartphones? *Rossiiskii tekhnologicheskii zhurnal = Russian Technological Journal*. 2021;9(2):22–34 (in Russ.). <https://doi.org/10.32362/2500-316X-2021-9-2-22-34>

Financial Disclosure: No author has a financial or property interest in any material or method mentioned.

The authors declare no conflicts of interest.

ВВЕДЕНИЕ

Пешеходная навигация занимает важное место среди всех навигационных задач: ее решение позволяет определять местоположение людей внутри больших зданий, таких как аэропорты, вокзалы, торговые центры и т.д. Решение такого рода задач особенно важно в случае чрезвычайных ситуаций – системы внутреннего позиционирования широко применяются пожарными расчетами и спасательными службами при взрывах и пожарах. Кроме того,

эти системы могут оказаться крайне полезными для людей с нарушениями зрительного аппарата.

Для решения таких задач могут использоваться различные источники информации: системы спутниковой навигации, сети Wi-Fi, Bluetooth и другие. Однако упомянутые источники доступны не во всех случаях, а соответствующие сигналы могут оказаться слишком слабыми (например, GPS-позиционирование дает плохую точность внутри помещений), что зачастую не позволяет построить решение исключительно на их основе. Поэтому для решения задач

пешеходной навигации также широко применяются инерциальные датчики, использование которых не требует какой-либо дополнительной инфраструктуры. На сегодняшний день такие датчики практически всегда устанавливаются в смартфонах и различных носимых устройствах, что позволяет создавать общедоступное решение задач позиционирования без применения каких-либо дополнительных средств.

Однако производители смартфонов используют датчики невысокой точности, что приводит к быстрому накоплению ошибок при использовании классических подходов, основанных на интегрировании кажущихся ускорений и угловых скоростей [1, 2]. Другим подходом к решению задач пешеходной навигации являются методы, сводящиеся к определению длины и направления каждого шага. При этом возникают дополнительные сложности, поскольку смартфон может использоваться в различных сценариях (например, он может находиться в руке, в кармане или в сумке), а его ориентация в пространстве может меняться независимо от направления движения. Существенными развитием данного подхода является использование методов машинного и глубокого обучения [2–4]. Однако методы машинного и глубокого обучения требуют наличия обучающих данных, содержащих наряду с информацией от датчиков смартфона достоверную информацию о реальных позициях пешехода. В работе [5] рассматриваются существующие датасеты, полученные различными способами, и представлен датасет «RuDaCoP», полученный при помощи закрепленных на ногах ИИБ. Целью данной работы является детальное описание алгоритмов, получающих такие референсные данные с помощью дешевых и простых в использовании инерциальных датчиков, а также оценка надежности этих алгоритмов.

При этом построение траекторий рассматривается не как самоцель, а как этап для построения навигационных алгоритмов для смартфонов. Для этого также решается задача синхронного сбора данных датчиками смартфонов, датчиками инерциальных блоков и высокоточными GNSS-приемниками, верифицирующими точностные характеристики разрабатываемых алгоритмов.

Предлагается метод восстановления траекторий по данным с инерциальных датчиков, закрепленных в области подъема стопы (рис. 1), используя алгоритмы фильтрации. Такой метод крепления позволяет использовать предположение о нулевой скорости устройств в моменты соприкосновения стопы с поверхностью и рассматривается в работах [6–8]. Для повышения точности восстановленных референсных траекторий вводится дополнительное ограничение – рассматриваются только замкнутые траектории, что не накладывает существенных ограничений на экспериментатора: требуется лишь вернуться в начальную

точку по окончании движения. Указанное предположение позволяет использовать информацию о совпадении начальной и конечной точек движения и значительно улучшить точность построенных траекторий при постобработке данных. Поскольку при использовании прямого фильтра Калмана новое измерение отражается лишь на участке траектории, соответствующей последнему шагу, предлагается алгоритм сглаживания, корректирующий всю вычисляемую кривую. Заметим, что ограничение на замкнутость рассматривается лишь применительно к обучающим данным – в реальных траекториях пешеход подобным условием не ограничен.



Рис. 1. Расположение измерительного блока на ноге

В случае, когда имеются данные с двух различных ИИБ, закрепленных соответственно на правой и левой ногах, можно дополнительно использовать предположение, что положения датчиков в пространстве не могут значительно отличаться друг от друга. Был разработан алгоритм, позволяющий использовать все вышеперечисленные данные (информация о начальной и конечной позициях, псевдонаблюдения скорости, информация о положении другой ноги) для построения сглаженных траекторий правой и левой ног, а затем построить по ним финальную обобщенную траекторию.

Работа алгоритмов была проверена на траекториях длительностью от 1 до 10 минут, проходящих на горизонтальных поверхностях. Для оценки качества результатов использовалось дополнительное GNSS-оборудование в RTK-режиме (особенности подхода подробно разобраны в разделе 5), позволяющее получить высокоточное навигационное решение. В работе приведены данные о максимальном отклонении RTK-решений от траекторий, восстановленных предложенными способами.

1. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ДВИЖЕНИЯ ИИБ

Динамическая система, описывающая движение чувствительного элемента, прикрепленного к ноге человека, является нелинейной и может быть записана в следующем виде:

$$\begin{cases} \mathbf{p}_n = \mathbf{p}_{n-1} + \mathbf{v}_{n-1} dt, \\ \mathbf{v}_n = \mathbf{v}_{n-1} + (\mathbf{C}_n^T \mathbf{f}_n + \mathbf{g}) dt, \\ \mathbf{C}_n = \mathbf{R}_n \mathbf{C}_{n-1}, \end{cases} \quad (1)$$

где n – индекс времени; dt – разница во времени между последовательными измерениями; $\mathbf{p}_n \in \mathbb{R}^3$ – координаты ИИБ в пространстве; $\mathbf{v}_n \in \mathbb{R}^3$ – скорость движения ИИБ; $\mathbf{f}_n \in \mathbb{R}^3$ и $\mathbf{w}_n \in \mathbb{R}^3$ – измерения идеальных акселерометра и гироскопа соответственно, \mathbf{g} – вектор силы тяжести; $\mathbf{C}_n \in \mathbb{R}^{3 \times 3}$ – матрица ориентации; $\mathbf{R}_n = \mathbf{R}_n(\mathbf{w}_n)$ – матрица поворота.

Интегрирование системы происходит на основе измерений, полученных с датчиков, при этом предполагается наличие немоделируемых ошибок измерений акселерометров и гироскопов:

$$\mathbf{f}_n = \tilde{\mathbf{f}}_n + \delta \mathbf{f}_n, \quad \mathbf{w}_n = \tilde{\mathbf{w}}_n + \delta \mathbf{w}_n.$$

Вычисленные координаты $\tilde{\mathbf{p}}_n$, скорость $\tilde{\mathbf{v}}_n$ и матрица ориентации $\tilde{\mathbf{C}}_n$ будут отличаться от истинных. Введем вектор ошибок $\delta \mathbf{x}_n = [\delta \mathbf{x}_n \ \delta \mathbf{v}_n \ \delta \mathbf{p}_n]^T$, который связывает истинные и вычисленные параметры:

$$\begin{cases} \delta \mathbf{p}_n = \mathbf{p}_n - \tilde{\mathbf{p}}_n, \\ \delta \mathbf{v}_n = \mathbf{v}_n - \tilde{\mathbf{v}}_n, \\ \mathbf{C}_n^T = (\mathbf{I}_{3 \times 3} + \hat{\mathbf{a}}_n) \tilde{\mathbf{C}}_n^T, \end{cases} \quad (2)$$

где $\mathbf{I}_{3 \times 3} \in \mathbb{R}^{3 \times 3}$ – единичная матрица, а крышкой обозначено преобразование трехмерного вектора в кососимметрическую матрицу:

$$\mathbf{a} \in \mathbb{R}^3, \quad \hat{\mathbf{a}} = \begin{bmatrix} 0 & a_3 & -a_2 \\ -a_3 & 0 & a_1 \\ a_2 & -a_1 & 0 \end{bmatrix} \in \mathbb{R}^{3 \times 3}.$$

Вектор \mathbf{p}_n называют углом малого поворота.

Для вектора ошибок $\delta \mathbf{x}_n$ справедливо линейное соотношение:

$$\delta \mathbf{x}_n = \mathbf{F}_n \delta \mathbf{x}_{n-1} + \mathbf{G}_n \begin{bmatrix} \delta \mathbf{f}_n \\ \delta \mathbf{w}_n \end{bmatrix}, \quad (3)$$

$$\mathbf{F}_n = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{I}_{3 \times 3} dt & \mathbf{O}_{3 \times 3} \\ \mathbf{O}_{3 \times 3} & \mathbf{I}_{3 \times 3} & -\widehat{\mathbf{C}}_n^T \mathbf{f}_n dt \\ \mathbf{O}_{3 \times 3} & \mathbf{O}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix},$$

$$\mathbf{G}_n = \begin{bmatrix} \mathbf{O}_{3 \times 3} & \mathbf{O}_{3 \times 3} \\ \tilde{\mathbf{C}}_n^T dt & \mathbf{O}_{3 \times 3} \\ \mathbf{O}_{3 \times 3} & -\tilde{\mathbf{C}}_n^T dt \end{bmatrix},$$

где $\mathbf{O}_{3 \times 3} \in \mathbb{R}^{3 \times 3}$ – нулевая матрица. Уравнение ошибок (3) выводится из определения (2) с учетом (1).

2. АЛГОРИТМ ВОССТАНОВЛЕНИЯ ТРАЕКТОРИИ ДВИЖЕНИЯ С ПОМОЩЬЮ ОДНОГО ИИБ

Алгоритм восстановления траекторий реализован в виде фильтра Калмана с обратной связью и сглаживающего RTS-фильтра. В качестве наблюдений используется информация о нулевой скорости ИИБ в моменты стационарного положения ноги при ходьбе, а также известные начальное и конечное положения. Алгоритм с использованием только псевдонаблюдений $\mathbf{v}_n = \mathbf{0}$ описан в [6].

Определение стационарного положения ноги основано на сравнении статистики T [9] с заданной константой $\gamma > 0$. Предполагается, что ИИБ находится в покое, если выполнено условие

$$T(\{\tilde{\mathbf{f}}_i, \tilde{\mathbf{w}}_i\}_{W_n}) < \gamma, \quad (4)$$

где $\{\tilde{\mathbf{f}}_i, \tilde{\mathbf{w}}_i\}_{W_n}$ – измерения акселерометра и гироскопа на временном интервале W_n .

В начале и в конце эксперимента ИИБ в течение нескольких секунд находится в положении покоя, поэтому в соответствующие моменты времени, определяемые на основе статистики T , можно использовать информацию о начальной позиции $\mathbf{p}_n = \mathbf{0}$ (без ограничения общности считаем ее нулевой).

Чтобы добавленная корректировка позиции по последним наблюдениям имела место не только в конце движения, но отражалась на всей траектории, использован сглаживающий RTS-фильтр, так же описанный в [6]. Кроме того, сглаживание позволяет избавиться от разрывов построенных траекторий, происходящих в ходе коррекции в конце каждого шага.

Поскольку система уравнений (1) нелинейна и накопление ошибки счисления может приводить к значительному расхождению с уравнениями ошибок (3), предлагается использовать фильтр Калмана с обратной связью по ориентации. Таким образом, значение матрицы ориентации корректируется на этапе прямого фильтра Калмана.

Псевдокод полученного алгоритма представлен в Алг. 1. Используются следующие обозначения: $\tilde{\mathbf{x}}_n$ соответствует набору вычисляемых значений $[\tilde{\mathbf{p}}_n, \tilde{\mathbf{v}}_n, \tilde{\mathbf{C}}_n]$; \mathbf{P}_n – ковариационная матрица вектора ошибок $\delta \mathbf{x}_n$; \mathbf{Q} – ковариационная матрица случайных ошибок измерений; \mathbf{R}' и \mathbf{R}'' – ковариационные матрицы наблюдений; функция f_{mech} соответствует формулам (1). Функция standstill в представленном

алгоритме возвращает истинные значения в моменты начального и конечного положения покоя с известной позицией.

```

Initialization:  $\mathbf{x}_0 \leftarrow E[\mathbf{x}_0]$ ,  $\delta\mathbf{x}_0 \leftarrow \mathbf{0}$ ,  $\mathbf{P}_0 \leftarrow \text{cov}(\mathbf{x}_0)$ 
/* Forward Kalman filter stage */
for  $n = 2$  to  $N$  do
     $\mathbf{x}_n \leftarrow f_{\text{mech}}(\mathbf{x}_{n-1}, \mathbf{f}_n, \mathbf{w}_n)$ 
     $\delta\mathbf{x}_{n|n-1} \leftarrow \mathbf{F}_n \delta\mathbf{x}_{n-1|n-1}$ 
     $\mathbf{P}_{n|n-1} \leftarrow \mathbf{F}_n \mathbf{P}_{n-1|n-1} \mathbf{F}_n^T + \mathbf{G}_n \mathbf{Q} \mathbf{G}_n^T$ 
    if  $T(\mathbf{f}_{W_n}, \mathbf{w}_{W_n}) < \gamma$  then
        if standstill( $n$ ) = true then
             $\mathbf{H} \leftarrow \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{O}_{3 \times 3} & \mathbf{O}_{3 \times 3} \\ \mathbf{O}_{3 \times 3} & \mathbf{I}_{3 \times 3} & \mathbf{O}_{3 \times 3} \end{bmatrix}$ 
             $\mathbf{K}_n \leftarrow \mathbf{P}_{n|n-1} \mathbf{H}^T (\mathbf{H} \mathbf{P}_{n|n-1} \mathbf{H}^T + \mathbf{R}')^{-1}$ 
             $\delta\mathbf{x}_{n|n} \leftarrow \delta\mathbf{x}_{n|n-1} + \mathbf{K}_n \begin{bmatrix} \delta\mathbf{p}_{n|n-1} + \mathbf{P}_n \\ \delta\mathbf{v}_{n|n-1} + \mathbf{v}_n \end{bmatrix}$ 
        else
             $\mathbf{H} \leftarrow \begin{bmatrix} \mathbf{O}_{3 \times 3} & \mathbf{I}_{3 \times 3} & \mathbf{O}_{3 \times 3} \end{bmatrix}$ 
             $\mathbf{K}_n \leftarrow \mathbf{P}_{n|n-1} \mathbf{H}^T (\mathbf{H} \mathbf{P}_{n|n-1} \mathbf{H}^T + \mathbf{R}'')^{-1}$ 
             $\delta\mathbf{x}_{n|n} \leftarrow \delta\mathbf{x}_{n|n-1} + \mathbf{K}_n (\delta\mathbf{v}_{n|n-1} + \mathbf{v}_n)$ 
        end
         $\mathbf{P}_{n|n} \leftarrow (\mathbf{I}_{9 \times 9} - \mathbf{K}_n \mathbf{H}) \mathbf{P}_{n|n-1}$ 
        /* Compensate internal angle states */
         $\mathbf{C}_n \leftarrow (\mathbf{I}_{3 \times 3} + \hat{\beta}_n) \mathbf{C}_n$ 
         $\hat{\beta}_n \leftarrow \mathbf{0}$ 
    end
end
/* Smoothing */
for  $n = N - 1$  to  $1$  do
     $\mathbf{A}_n \leftarrow \mathbf{P}_{n|n} \mathbf{F}_n^T \mathbf{P}_{n+1|n}^{-1}$ 
     $\delta\mathbf{x}_{n|N} \leftarrow \delta\mathbf{x}_{n|n} + \mathbf{A}_n (\delta\mathbf{x}_{n+1|N} - \delta\mathbf{x}_{n+1|n})$ 
     $\mathbf{P}_{n|N} \leftarrow \mathbf{P}_{n|n} + \mathbf{A}_n (\mathbf{P}_{n+1|N} - \mathbf{P}_{n+1|n}) \mathbf{A}_n^T$ 
end
/* Compensate internal states */
for  $n = 1$  to  $N$  do
     $\mathbf{x}_n \leftarrow \mathbf{x}_n + \delta\mathbf{x}_{n|N}$ 
     $\delta\mathbf{x}_n \leftarrow \mathbf{0}$ 
end

```

Алг. 1. Получение навигационного решения с учетом замкнутости траекторий движения

3. МЕТРИКА ПОДОБИЯ ТРАЕКТОРИЙ

Для первичной оценки правдоподобия результатов предложенного алгоритма можно провести эксперимент, в котором ИИБ закреплены на обеих ногах человека, а затем сравнить две восстановленные траектории между собой. Для этого необходима некоторая метрика, позволяющая сравнивать две кривые. Эта же метрика понадобится, чтобы оптимальным образом наложить две траектории друг на друга (начальные ориентации ИИБ не совпадают). При этом метрика должна учитывать следующие особенности: длительности эксперимента для левой и правой ног, вообще говоря, различны (датчики включаются неодновременно), ИИБ работают несинхронно, возможны пропуски измерений.

В работе [10] были рассмотрены две метрики, удовлетворяющие заданным требованиям: дискретное расстояние Фреше и DTW-метрика. Наиболее простые алгоритмы [11, 12] позволяют с помощью метода динамического программирования вычислить указанные метрики за полиномиальное время. Согласно [10], для достижения более

точных результатов в качестве метрики подобия траекторий следует использовать DTW-метрику, вычисляемую на основе модифицированного алгоритма. Для этого вводится дополнительное ограничение на максимально допустимую разницу между индексами соответствующих друг другу точек (под индексом понимается порядковый номер измерения в эксперименте). Такой подход позволяет не только повысить точность сопоставления точек кривых друг другу, но также уменьшить время вычислений.

Помимо значения метрики, результатом работы модифицированного алгоритма является последовательность пар сопоставленных друг другу точек траекторий левой и правой ног (некоторые из этих пар обозначены красным цветом на рис. 2). Если для каждой такой пары точек взять точку, лежащую посередине соединяющего их отрезка, получится некоторая усредненная кривая. Анализ такого подхода выделения «средней» траектории с помощью DTW-алгоритма представлен в [13]. Отметим, что при усреднении траекторий таким способом теряется информация о времени измерений, поскольку точки, сопоставленные друг другу, могут быть получены в разные моменты времени.

Если информация о времени важна (например, она может понадобиться для сравнения результатов алгоритма с известным высокоточным решением), то для усреднения двух траекторий достаточно проинтерполировать точки одной из них для приведения позиций разных ног к одной временной сетке. При этом середина отрезка, соединяющего положение ног в один момент времени, будет примерно соответствовать положению центра масс человека.

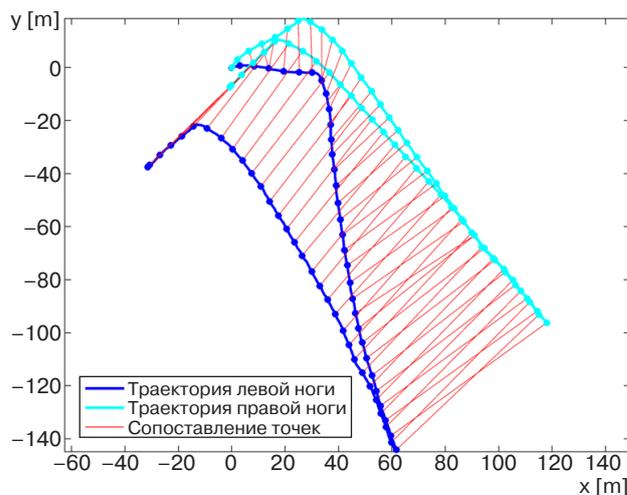


Рис. 2. Проекция траекторий левой и правой ног на горизонтальную плоскость и сопоставление точек восстановленных траекторий с помощью модифицированного DTW-алгоритма

4. АЛГОРИТМ ВОССТАНОВЛЕНИЯ ТРАЕКТОРИЙ ДВИЖЕНИЯ НА ОСНОВЕ ДВУХ ИИБ

Идея использования двух устройств заключается в том, что в моменты ходьбы, когда одна нога покоится, а вторая находится в «середине» шага, положения датчиков в проекции на горизонтальную плоскость должны находиться на расстоянии нескольких сантиметров друг от друга. Этот факт подробно исследован в [14]. Потому можно одновременно строить обе траектории, но для каждого ИИБ в качестве дополнительных наблюдений использовать позицию второго устройства.

Предлагаемый алгоритм можно декомпозировать следующим образом:

1. Построение каждой из двух траекторий с помощью Алг. 1 и определение начального несоответствия ориентаций ИИБ. Для этого путем подбора начальных углов минимизируется значение DTW-метрики.
2. Определение ноги, которой был сделан первый шаг, на основе условия (4) и имеющихся измерений.
3. Параллельный расчет траекторий по измерениям двух ИИБ с использованием наблюдений о нулевой скорости и положении другой ноги.

```

cur ← 2 // begin computation with the second leg
pos ←  $\tilde{x}_{k_1}^1$  // current position of the first leg
while  $k_1 < N_1 \parallel k_2 < N_2$  do
  /* select data for the current leg */
   $\tilde{x}_{n-1} \leftarrow \tilde{x}_{n-1}^{cur}, \delta\tilde{x}_{n-1} \leftarrow \delta\tilde{x}_{n-1}^{cur}, k \leftarrow k^{cur}, \tilde{f}_n \leftarrow \tilde{f}_n^{cur}, \tilde{w}_n \leftarrow \tilde{w}_n^{cur}$ 

  step_len = count_step_length(k, { $\tilde{f}_i, \tilde{w}_i$ })
  for n = k + 1 to k + step_len do
    [ $\tilde{x}_n, \delta\tilde{x}_{n|n-1}, P_{n|n-1}$ ] ← predict( $\tilde{x}_{n-1}, \delta\tilde{x}_{n-1}, \tilde{f}_n, \tilde{w}_n$ )
    if n = k + step_len / 2 then
      H ← [ $I_{2 \times 2} \quad O_{2 \times 7}$ ]
       $K_n \leftarrow P_{n|n-1} H^T (H P_{n|n-1} H^T + R^m)^{-1}$ 
       $\tilde{x}_n^{1,2} \leftarrow H \tilde{x}_n, \delta\tilde{x}_{n|n-1}^{1,2} \leftarrow H \delta\tilde{x}_{n|n-1}$  // select 2 coordinates
       $\delta\tilde{x}_{n|n} \leftarrow \delta\tilde{x}_{n|n-1} + K_n (pos + \delta\tilde{x}_{n|n-1}^{1,2} + \tilde{x}_n^{1,2})$ 
       $P_{n|n} \leftarrow (I_{9 \times 9} - K_n H) P_{n|n-1}$ 
    end
  end
  n ← k + step_len + 1

  while  $T(\{\tilde{f}_i, \tilde{w}_i\}_{W_n}) < \gamma$  do
    [ $\tilde{x}_n, \delta\tilde{x}_{n|n-1}, P_{n|n-1}$ ] ← predict( $\tilde{x}_{n-1}, \tilde{f}_n, \tilde{w}_n$ )
    if standstill(n) = true then
      H ← [ $I_{3 \times 3} \quad O_{3 \times 3} \quad O_{3 \times 3}$ ], R ← R'
    else
      H ← [ $O_{3 \times 3} \quad I_{3 \times 3} \quad O_{3 \times 3}$ ], R ← R''
    end
    [ $\delta\tilde{x}_{n|n}, P_{n|n}$ ] ← correct(H, R, P_{n|n-1},  $\tilde{x}_n, \delta\tilde{x}_{n|n-1}$ )
    n ← n + 1
  end
  pos ←  $\tilde{x}_n^{1,2}$  // save current position for the following observation

   $\tilde{x}_n^{cur} \leftarrow \tilde{x}_n, \delta\tilde{x}_n^{cur} \leftarrow \delta\tilde{x}_{n|n}, P_{n|n}^{cur} \leftarrow P_{n|n}, k^{cur} \leftarrow n$ 
  cur ← (3 - cur)
end

```

Алг. 2. Алгоритм получения замкнутого навигационного решения, использующий данные с двух ИИБ

4. Сглаживание траекторий RTS-фильтром, аналогичное таковому в Алг. 1.
5. Построение одной усредненной траектории.

Пункт 3 указанной декомпозиции более подробно представлен в Алг. 2 (переменные с верхним индексом 1 соответствуют датчику, который прикреплен к ноге, делающей первый шаг; переменные с верхним индексом 2 – другому датчику). Функции predict и correct соответствуют шагам прогноза и коррекции фильтра Калмана (см. Алг. 1).

5. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

А. Описание экспериментов

Для верификации предлагаемого подхода была выбрана следующая методология. В процессе эксперимента к ногам испытуемого с помощью специальных эластичных креплений монтировались два ИИБ. Также у испытуемого находился смартфон, получающий и записывающий в режиме реального времени данные от ИИБ. Наконец, для построения высокоточных референсных траекторий испытуемый был оснащен GNSS-приемником, работающим в RTK-режиме. Переносная GNSS-база была смонтирована на расстоянии, не превышающем 500 м от рабочей зоны эксперимента. Рис. 3 содержит визуальное пояснение расположения оборудования.

С инженерной точки зрения для осуществления верификации необходимо решить несколько задач: 1) беспроводной сбор сенсорных данных с закрепленных на ногах инерциальных блоков; 2) синхронизация во времени собранных данных от инерциальных блоков с учетом слабой точности внутренних тактовых генераторов; 3) получение высокоточного референсного RTK-решения от GNSS приемника; 4) синхронизация и сравнение референсного RTK-решения и навигационного решения, полученного с помощью предлагаемых алгоритмов.

Для сбора инерциальных данных с ног испытуемого были использованы модули MIMU22BLPX¹ (рис. 3г), в которых в качестве сенсорных элементов задействованы измерительные микросхемы Invensense MPU-9250². Каждый из модулей содержит массив из четырех Invensense MPU-9250, расположение и ориентация которых позволяет минимизировать систематические ошибки. Каждая из микросхем Invensense MPU-9250 включает в себя

¹ URL: http://www.inertiaelements.com/%20documents/mimu22bl/MIMU22BL_product-brief.pdf. [URL: http://www.inertiaelements.com/%20documents/mimu22bl/MIMU22BL_product-brief.pdf]

² URL: <https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf>. [URL: <https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf>]



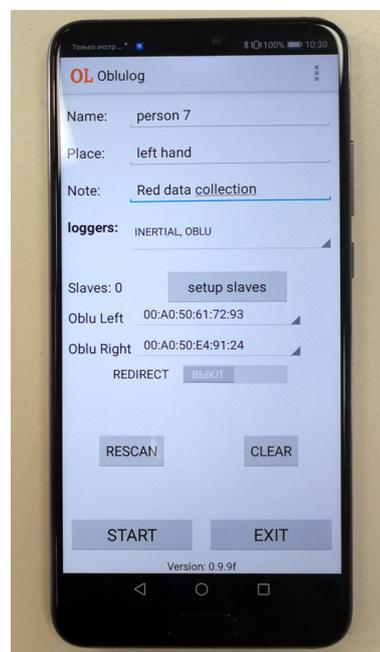
Рис. 3. Визуальное пояснение расположения оборудования: (а) экспериментатор, оснащенный закрепленной на шлеме антенной GNSS приемника, смартфоном для записи данных и двумя ИИБ на ногах (за пределами кадра); (б) ИИБ, прикрепленный на ногу экспериментатора; (в) базовая станция GNSS для получения референсного RTK-решения; (г) ИИБ MIMU22BLPX крупным планом

трехосевые акселерометры, гироскопы и магнетометры. Блоки MIMU22BLPX, помимо измерительных микросхем, содержат приемо-передатчик Bluetooth Low Energy и аккумулятор, позволяющий работать автономно в течение достаточного времени. Массогабаритные характеристики блоков определяются размером $42.2 \times 27.9 \times 17.0$ мм и весом в 20 г. Использование беспроводного интерфейса Bluetooth LE позволяет получать данные от блоков с частотой 125 Hz. ИИБ крепятся к ногам с помощью эластичного крепления, гарантирующего неподвижность блоков относительно голеностопа испытуемого, достаточную для получения качественных данных.

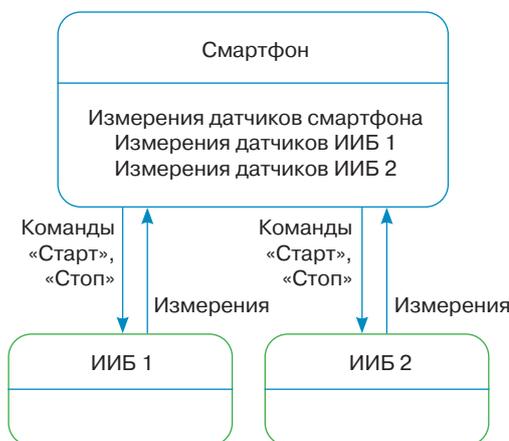
Для записи измерений блоков было разработано программное обеспечение (ПО) для смартфона, позволяющее записывать данные измерительных блоков в беспроводном режиме. ПО содержит интерфейс управления для старта и остановки записи данных ИИБ. Во время эксперимента данные в режиме реального времени передаются на управляющий смартфон и накапливаются в нем. Внешний вид интерфейса ПО отражен на рис. 4а. Общая логика работы программно-аппаратного комплекса проиллюстрирована на рис. 4б. Местоположение смартфона во время записи данных не принципиально. Также ПО позволяет одновременно записывать данные от инерциальных датчиков нескольких смартфонов (датасет, содержащий синхронизированные данные нескольких смартфонов и двух ИИБ на ногах, подробно представлен в работе [5]). Пост-обработка данных от ИИБ, включающая временную синхронизацию блоков и построение единого навигационного решения, осуществляется позже на вычислительной машине.

Б. Синхронизация устройств

Каждый пакет с данными сенсоров от обоих ИИБ содержит информацию о времени. Однако использование относительно недорогих ИИБ имеет существенный недостаток, связанный с низкой точностью



(а)



(б)

Рис. 4. Аппаратно-программное обеспечение для сбора данных: (а) интерфейс ПО для смартфона; (б) общая логика работы аппаратно-программного комплекса

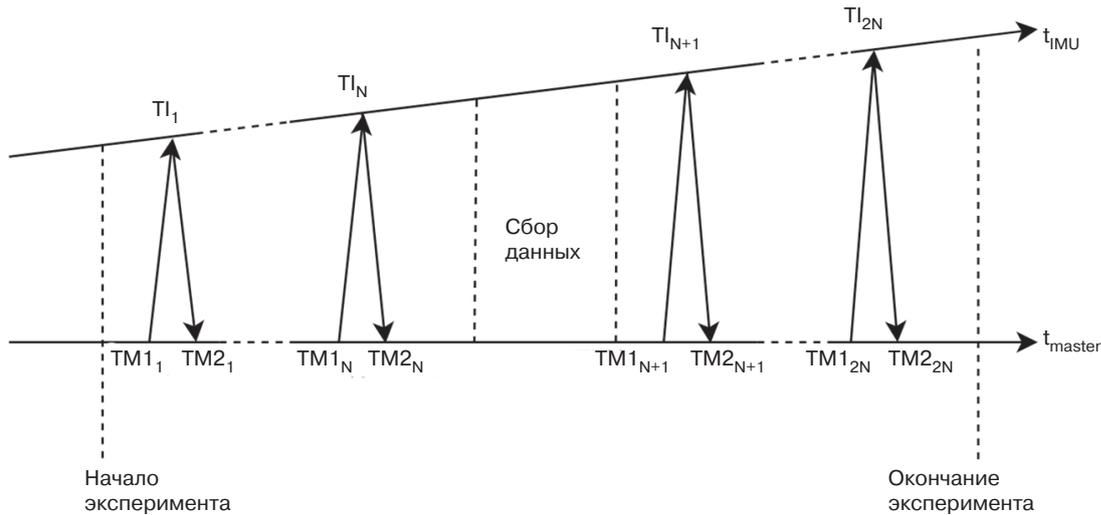


Рис. 5. Временная диаграмма обмена синхронизационными данными управляющего смартфона и инерциального модуля во время эксперимента

внутренних часов. Причем частота внутреннего тактового генератора меняется в ходе эксперимента, поэтому предварительной временной калибровки блоков недостаточно. Решением проблемы синхронизации могла бы оказаться фиксация времени получения пакетов от ИИБ на смартфоне. Но и этот вариант не работает из-за буферизации процесса передачи данных через беспроводной канал. Поэтому необходим дополнительный алгоритм синхронизации измерений датчиков в режиме пост-обработки.

Предлагаемый подход основывается на предположении о том, что зависимость времени ИИБ от времени управляющего смартфона может быть линейно аппроксимирована с достаточной точностью. Таким образом, необходимо вычислить коэффициент масштабирования и коэффициент сдвига b . Для этого в начале и в конце эксперимента осуществляются несколько сеансов обмена сообщениями, помеченными временными метками, согласно рис. 5. Каждый такой сеанс описывается вектором вида $[TM1_i, TI_i, TM2_i]$, где $TM1_i$ – время отправки сообщения от управляющего смартфона к измерительному блоку (здесь и далее, i – номер сеанса); TI_i – время фиксации полученного сообщения во временной шкале ИИБ и время отправления нового сообщения к смартфону; $TM2_i$ – время получения смартфоном последнего сообщения.

Модель синхронизации основана на работе [15], при этом в алгоритм вводятся модификации, позволяющие учесть особенности аппаратной системы. Исходя из предположения о линейной зависимости времени ИИБ от времени управляющего смартфона, для всех сеансов обмена сообщениями, представленных на рис. 5, будет справедливо следующее соотношение:

$$TM1_i < aTI_i + b < TM2_i,$$

Это позволяет определить интервалы значений, которым принадлежат a и b , путем решения задач линейного программирования LP1 и LP2:

$$LP1: \begin{cases} c^T x \rightarrow \min, \\ Ax < B, \end{cases} \quad LP2: \begin{cases} c^T x \rightarrow \max, \\ Ax < B, \end{cases}$$

где введены обозначения:

$$x = \begin{bmatrix} a \\ b \end{bmatrix}, \quad c = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad A = \begin{bmatrix} TI_1 & 1 \\ -TI_1 & -1 \\ \dots & \dots \\ TI_{2N} & 1 \\ -TI_{2N} & -1 \end{bmatrix}, \quad B = \begin{bmatrix} TM2_1 \\ -TM1_1 \\ \dots \\ TM2_{2N} \\ -TM1_{2N} \end{bmatrix}.$$

Если вектор $x_1 = [a_1, b_1]^T$ является решением задачи LP1, а $x_2 = [a_2, b_2]^T$ – решение задачи LP2, то интервалы значений, которым принадлежат a и b , имеют следующий вид:

$$a_1 \leq a \leq a_2,$$

$$\min(b_1, b_2) \leq b \leq \max(b_1, b_2),$$

и в качестве коэффициентов синхронизации можно выбрать:

$$a = \frac{a_1 + a_2}{2}, \quad b = \frac{b_1 + b_2}{2}.$$

Решения задач линейного программирования, с учетом сделанных допущений, можно получить, например, с помощью симплекс-метода [16].

При составлении закона преобразования времени ИИБ во время управляющего смартфона, следует

учесть, что Bluetooth LE вносит задержку в прием и передачу сигнала, определяемую экспериментально. На практике такую задержку можно считать постоянной. В тестируемой системе данная задержка составила $\Delta_{ble} \approx 0.03$ с. Таким образом, для синхронизации было использовано соотношение:

$$t_I^M = at_I^I + b - \Delta_{ble},$$

где I – целочисленная неоднозначность фазовых измерений; t_I^M – время ИИБ во временной шкале управляющего смартфона; t_I^I – время ИИБ в собственной временной шкале.

Определение текущего времени на смартфоне, работающем под управлением ОС Android, осуществлялось при помощи метода *SystemClock.elapsedRealtime()*. Данный метод возвращает количество миллисекунд, прошедших после включения телефона³. При использовании данного метода исключается возможность влияния алгоритмов синхронизации, работающих на уровне ОС. Определение времени на ИИБ осуществлялось при помощи запроса текущего времени модуля.

Точность работы алгоритма синхронизации определялась путем проведения серии экспериментов, в ходе которых управляющему телефону и ИИБ придавалось одновременное вращение, после чего проводилось сопоставление показаний их гироскопов. В результате было определено, что рассогласование по времени между данными ИИБ и данными управляющего смартфона не превышает 10 мс на десятиминутной траектории.

В. Референсное RTK-решение

Поскольку рассматриваемый в работе подход предполагает точность в единицы метров, референсные данные для его верификации должны быть точнее на 1–2 порядка. В этой связи использование типového GNSS-приемника является неудовлетворительным решением. Хорошо известно, что определение положения при помощи спутниковых навигационных систем основано на измерении времени распространения радиосигнала от нескольких спутников до приемника. При этом выделяют два типа измерений: 1) кодовые, фиксирующие смещение псевдослучайной последовательности (псевдодалности) pr , которой модулирован несущий сигнал; 2) фазовые, фиксирующие смещение фазы ph несущей радиосигнала. Модели этих измерений имеют следующий вид:

$$pr = \rho(x_{sv}, x_{rcv}) + c(\tau_{sv} + \tau_{rcv}) + \Delta_{ion} + \Delta_{trop} + r_{pr},$$

$$ph = \frac{1}{\lambda} \rho(x_{sv}, x_{rcv}) + \frac{c}{\lambda} (\tau_{sv} + \tau_{rcv}) - \frac{1}{\lambda} \Delta_{ion} + \frac{1}{\lambda} \Delta_{trop} + I + r_{ph},$$

где x_{rcv} – координаты приемника; x_{sv} – координаты навигационного спутника; $\rho(x_{sv}, x_{rcv})$ – геометрическое расстояние между спутником и приемником; τ_{rcv} – ошибка часов приемника; τ_{sv} – ошибка часов спутника; Δ_{ion} – задержка сигнала в ионосфере Земли; Δ_{trop} – задержка сигнала в тропосфере планеты; r_{ph} и r_{pr} – немоделируемые (случайные) ошибки измерений фазы и псевдодалности соответственно; I – целочисленная неоднозначность фазовых измерений; c – скорость света в вакууме; λ – длина волны несущего радиосигнала.

Для типового подхода в решении задачи навигации по измерениям псевдодалностей оцениваемыми параметрами являются координаты и ошибка часов приемника. При наличии измерений от четырех и более спутников в один момент времени решается задача оптимизации:

$$[\tilde{x}_{rcv}, \tilde{\tau}_{rcv}] = \arg \min_{x, \tau} \sum_j (z_{pr}^j - \rho(x_{sv}^j, x) - c\tau)^2,$$

где $z_{pr}^j = pr^j - c\tau_{sv}^j - \Delta_{ion}^j - \Delta_{trop}^j$; индекс j обозначает номер спутника. Точность определения параметров составляет от единиц до десятков метров. Это связано с точностью измерения псевдодалностей, ошибкой моделирования дополнительных слагаемых и вычислением координат спутников по эфемеридной информации. Подобная точность для референсного решения является недостаточной.

Точность фазовых измерений значительно выше, чем у псевдодалностей (до нескольких миллиметров), но их неоднозначность I не позволяет полностью реализовать этот потенциал при оценке координат в стандартном режиме. Однако ошибки параметров спутника (x_{sv}, τ_{sv}) медленно меняются во времени и их можно считать одинаковыми для всех приемников. Задержки сигнала в атмосфере Земли зависят от пути распространения сигнала и сильно коррелируют для приемников на небольшом расстоянии друг от друга (до нескольких километров). Поэтому при использовании двух приемников, можно достичь более высокой точности определения их относительного положения в реальном времени. Для этого используются двойные разности измерений фазы и псевдодалностей, в которых отсутствуют дополнительные слагаемые, кроме фазовых неоднозначностей:

³ URL: [https://developer.android.com/reference/android/os/SystemClock.html#elapsedRealtime\(\)](https://developer.android.com/reference/android/os/SystemClock.html#elapsedRealtime()). [URL: http://www.inertialelements.com/%20documents/mimu22bl/MIMU22BL_product-brief.pdf]

$$\Delta\Delta ph = \frac{1}{\lambda} \left[\rho(x_{sv}^j, x_{rov}) - \rho(x_{sv}^0, x_{rov}) - \rho(x_{sv}^j, x_{base}) + \rho(x_{sv}^0, x_{base}) \right] + \Delta\Delta I + \Delta\Delta r_{ph}, \quad (5)$$

$$\Delta\Delta pr = \rho(x_{sv}^j, x_{rov}) - \rho(x_{sv}^0, x_{rov}) - \rho(x_{sv}^j, x_{base}) + \rho(x_{sv}^0, x_{base}) + \Delta\Delta r_{pr}, \quad (6)$$

где $\Delta\Delta r_{ph}$, $\Delta\Delta r_{pr}$ – немоделируемые ошибки двойных разностей, которые включают в себя остаточные ошибки от ионосферных и тропосферных задержек и от вычисления параметров спутников. Двойные разности составляются между измерениями от двух спутников и двух приемников (неподвижной базой и ровером). Обычно выбирается один ведущий спутник, и его измерения вычитаются из остальных. Подобный подход к построению навигационного решения часто называют RTK (Real Time Kinematic). В RTK-решении применяется фильтр для оценки параметров, которые включают в себя неоднозначности фазовых измерений ($\Delta\Delta I$), после чего на основе измерений (5)–(6) восстанавливается относительное положение ровера ($x_{rov} - x_{base}$).

Для получения качественных референсных траекторий эксперименты проводились под открытым небом вдали от высоких построек. Использовались два 72-канальных одночастотных GNSS приемника Emlid Reach M+⁴, которые позволяют записывать измерения псевдодальностей и фаз, а также навигационные сообщения от спутников с частотой 5 Гц. Использовались активные антенны, экранированные снизу радионепроницаемой пластиной, для исключения приема переотраженного сигнала. Базовый GNSS-приемник был закреплен на неподвижной треноге в течение всего эксперимента. Второй GNSS-приемник (подвижный) был закреплен на шлеме экспериментатора. После окончания эксперимента

⁴ URL: <https://emlid.com/reach/>. [URL: <https://emlid.com/reach/>]

наборы данных от базового и подвижного приемников совместно обрабатывались с помощью программы *rtkexplorer*⁵, которая формирует окончательное навигационное решение. Заявленная точность подобного решения составляет порядка 1 см в горизонтальной плоскости.

6. РЕЗУЛЬТАТЫ

На рис. 6 показаны траектории, полученные с помощью фильтра Калмана [6] (без учета замкнутости траекторий, но с использованием сглаживания), а также траектории, полученные с помощью Алг. 1 и Алг. 2 (зеленые траектории соответствуют левой ноге, синие – правой, красным цветом обозначена усредненная траектория). Кроме того, на рис. 6г сравниваются полученная усредненная траектория и RTK-решение (обозначено фиолетовым цветом). Использование предложенных алгоритмов позволяет существенно скорректировать восстановленные траектории.

В случае, когда доступен только один ИИБ, для восстановления траекторий предлагается использовать Алг. 1, однако для оценки его результатов не представляется возможным с достаточной точностью вычислить отклонение построенных решений от реальных траекторий. Это связано с тем, что во время ходьбы положение отдельной ноги, в отличие от центра масс человека, вообще говоря, не совпадает с положением RTK-приемника. Тем не менее, оценками могут служить DTW-расстояния между независимо построенными траекториями левой и правой ног. В таблице 1 представлены DTW-расстояния, полученные без учета и с учетом замкнутости траекторий.

В таблице 2 представлены максимальные отклонения результатов Алг. 2 от соответствующих им RTK-решений (под отклонением понимается евклидово расстояние между точками траекторий

⁵ URL: <http://rtkexplorer.com/>. [URL: <http://rtkexplorer.com/>]

Таблица 1. Результаты работы алгоритмов, использующих один ИИБ

N	Длительность эксперимента, с	Длина траектории, м	DTW-расстояние в результате Алг. [6], м	DTW-расстояние в результате Алг. 1, м
1	88.3	71.92	0.45	0.09
2	200.9	218.45	3.57	0.64
3	203.0	225.54	1.04	0.44
4	215.0	240.39	4.93	0.50
5	321.0	382.09	1.89	2.07
6	342.1	373.05	16.27	2.21
7	625.5	706.68	13.96	5.49

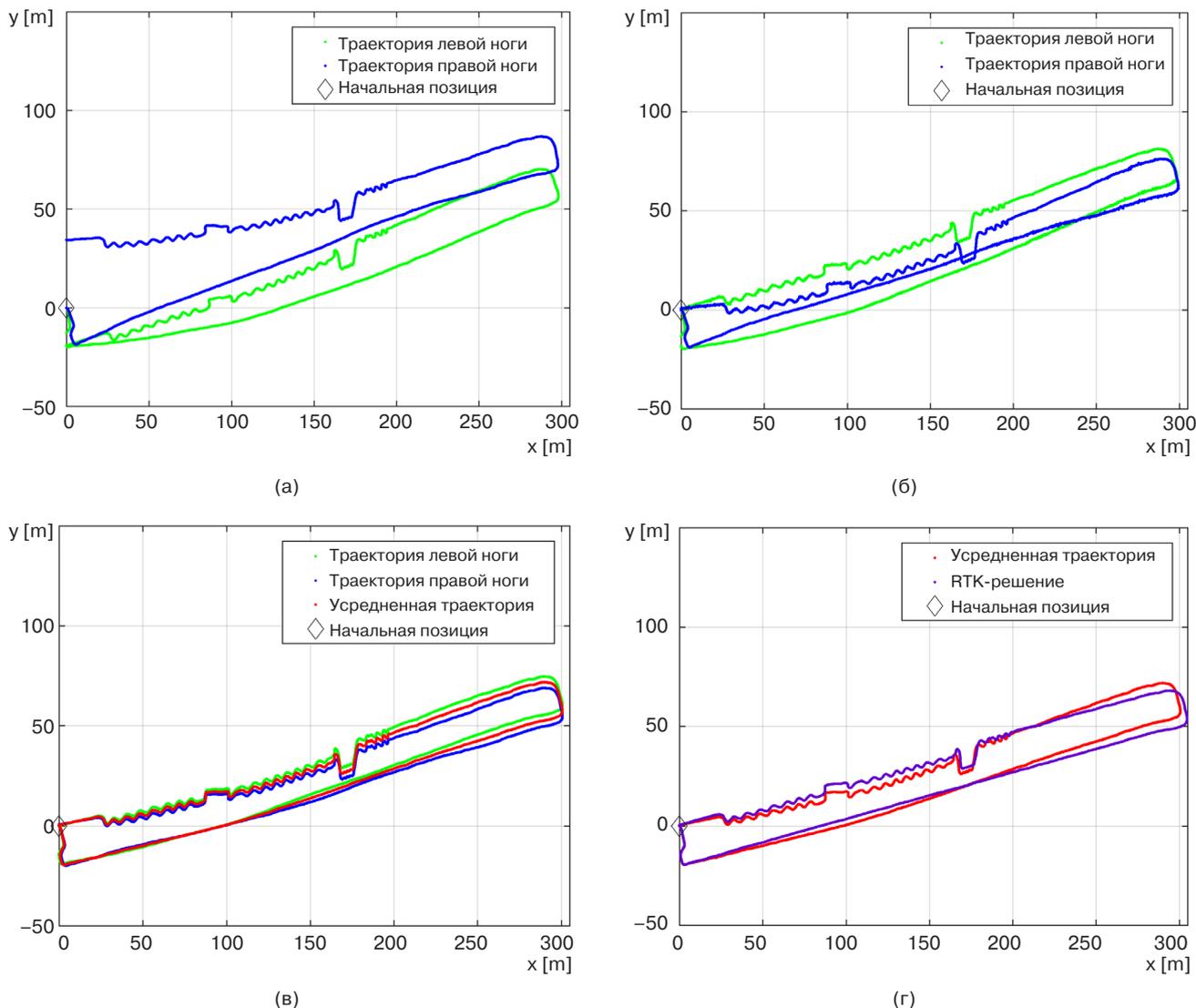


Рис. 6. Сравнение результатов алгоритмов на одном и том же эксперименте:

- (а) алгоритм [6] – траектории левой и правой ног восстанавливаются независимо друг от друга;
- (б) алг. 1 – траектории восстанавливаются независимо, но с учетом их замкнутости; (в) алг. 2 – одновременно используются оба ИИБ, учитывается замкнутость; (г) сравнение результата Алг. 2 с RTK-решением

в фиксированный момент времени) на экспериментальном наборе траекторий, а также для каждого эксперимента указано максимальное отношение от-

клонения к длине пройденного пути. В приведенных примерах величины максимального отклонения составляют не более 2% от полных длин траекторий.

Таблица 2. Результаты работы алгоритма 2, использующего два ИИБ

N	Длительность эксперимента, с	Длина траектории, м	Максимальное отклонение, м	Макс. отношение отклонения к длине пройденного пути
1	88.3	71.92	0.53	0.03
2	200.9	218.45	2.08	0.06
3	203.0	225.54	1.15	0.02
4	215.0	240.39	3.94	0.04
5	321.0	382.09	6.57	0.09
6	342.1	373.05	3.16	0.09
7	625.5	706.68	6.01	0.11

ЗАКЛЮЧЕНИЕ

В работе представлены алгоритмы для восстановления замкнутых траекторий движения пешехода на основе информации об ускорениях и угловых скоростях (для одного и для двух инерциальных измерительных блоков, прикрепленных к ногам), приведен соответствующий псевдокод. Предложены два способа сравнения полученных траекторий, рассмотрены их достоинства и недостатки, а также указан способ оптимизации времени вычислений. Кроме того, предлагается способ построения одной усредненной траектории движения человека по двум траекториям движения инерциальных измерительных блоков, закрепленных на каждой из его ног. Для верификации результатов разработан программно-аппаратный

комплекс, позволяющий одновременно записывать данные инерциальных модулей, сенсоров смартфона и высокоточных GNSS-приемников, работающих в RTK-режиме. Рассмотрены задачи синхронизации оборудования. Предложенные алгоритмы продемонстрировали достаточную точность на экспериментальных данных, собранных с помощью дешевого и удобного оборудования, что позволяет использовать данный подход к построению референсных траекторий для решения задач пешеходной навигации для смартфонов как классическими методами, так и методами машинного и глубокого обучения.

Вклад авторов. Все авторы в равной степени внесли свой вклад в исследовательскую работу.

Authors' contribution. All authors equally contributed to the research work.

СПИСОК ЛИТЕРАТУРЫ / REFERENCES

1. Woodman O.J. An introduction to inertial navigation. *Technical Report UCAM-CL-TR-696*. 2007. 37 p. URL: <https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-696.pdf>
2. Chen C., Zhao P., Lu C.X., Wang W., Markham A., Trigoni N. Deep-Learning-Based Pedestrian Inertial Navigation: Methods, Data Set, and On-Device Inference. *IEEE Internet of Things Journal*. 2020;7(5):4431–4441.
3. Herath S., Yan H., Furukawa Y. RoNIN: Robust Neural Inertial Navigation in the Wild: Benchmark, Evaluations, & New Methods. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 2020, p. 3146–3152.
4. Wang Q., Luo H., Ye L., Men A., Zha, F., Huang, Y., Ou C. Pedestrian heading estimation based on spatial transformer networks and hierarchical LSTM. *IEEE Access*. 2019;7:162309–162322. <https://doi.org/10.1109/ACCESS.2019.2950728>
5. Bayev A., Chistyakov I., Derevyankin A., Gartseev I., Nikulin A., Pikhletsky M. RuDaCoP: The dataset for smartphone-based intellectual pedestrian navigation. In: *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Pisa, Italy, 2019, p. 1–8. <https://doi.org/10.1109/IPIN.2019.8911823>
6. Colomar D.S., Nilsson J.-O., Händel P. Smoothing for ZUPT-aided INSs. In: *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Sydney, NSW, Australia, 2012, p. 1–5. <https://doi.org/10.1109/IPIN.2012.6418869>
7. Nilsson J.-O., Skog I., Händel P., Hari K.V.S. Foot-mounted INS for everybody — an open-source embedded implementation. In: *Proceedings of the 2012 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. Myrtle Beach, SC, USA, 2012, p. 140–145. <https://doi.org/10.1109/PLANS.2012.6236875>
8. Nilsson J.-O., Gupta A.K., Händel P. Foot-mounted inertial navigation made easy. In: *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. 2014, p. 24–29.
9. Skog I., Handel P., Nilsson J.-O., Rantakokko J. Zero-velocity detection: An algorithm evaluation. *IEEE Transactions on Biomedical Engineering*. 2010;57(11):2657–2666. <https://doi.org/10.1109/TBME.2010.2060723>
10. Chistyakov I.A., Nikulin A.A., Gartseev I.B. Pedestrian dead-reckoning algorithms for dual foot-mounted inertial sensors. In: *2019 26th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*. St. Petersburg, Russia, 2019, p. 1–8. <https://doi.org/10.23919/ICINS.2019.8769341>
11. Eiter T., Mannila H. Computing discrete Fréchet distance. *Technical Report CD-TR 94/64*. Christian Doppler Laboratory for Expert Systems. 1994:636–637.
12. Müller M. Dynamic Time Warping. In: *Information Retrieval for Music and Motion*. Berlin, Heidelberg: Springer; 2007. P. 69–84. https://doi.org/10.1007/978-3-540-74048-3_4
13. Vaughana N., Gabrysa B. Comparing and combining time series trajectories using Dynamic Time Warping. *Procedia Computer Science*. 2016;96:465–474. <https://doi.org/10.1016/j.procs.2016.08.106>
14. Niu X., Li Y., Kuang J., Zhang P. Data fusion of dual foot-mounted IMU for pedestrian navigation. *IEEE Sensors Journal*. 2019;19(12):4577–4584. <https://doi.org/10.1109/JSEN.2019.2902422>
15. Lemmon M.D., Ganguly J., Xia L. Model-based clock synchronization in networks with drifting clocks. In: *Proceedings of Pacific Rim International Symposium on Dependable Computing*. Los Angeles, CA, USA, 2000, p. 177–184. <https://doi.org/10.1109/PRDC.2000.897300>
16. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. *Introduction to Algorithms*: 3rd Edition. The MIT Press; 2009. P. 864–878.

Об авторах

Чистяков Иван Александрович, аспирант кафедры системного анализа факультета вычислительной математики и кибернетики Московского государственного университета им. М.В. Ломоносова (119234, Россия, Москва, Ленинские горы, д. 1, стр. 52). E-mail: chistyakov.ivan@yahoo.com. Scopus Author ID: 57212444724, <https://orcid.org/0000-0002-1029-3690>

Гришов Иван Владимирович, аспирант кафедры «Проблемы управления» Института кибернетики ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78). E-mail: ivan.grishov@yandex.ru.

Никулин Алексей Андреевич, научный сотрудник ООО «КС Кадровый консалтинг» (115191, Москва, Большой Староданиловский пер., д. 2, стр. 7). E-mail: nikulin.alexey@huawei.com.

Пихлецкий Михаил Викторович, к.т.н., ведущий исследователь ООО «Техкомпания Хуавэй» (121614, Россия, Москва, Алтуфьевское шоссе, д. 1/7). E-mail: pikhletsky.mikhail@huawei.com.

Гарцев Илья Борисович, к.т.н., доцент кафедры «Проблемы управления» Института кибернетики ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78); ведущий инженер ключевых проектов ООО «Техкомпания Хуавэй» (121614, Россия, Москва). E-mail: gartseev@mirea.ru. ResearcherID: Y-6501-2019, Scopus Author ID: 55973474600, <https://orcid.org/0000-0001-5866-3411>

About the authors

Ivan A. Chistyakov, Postgraduate Student, System Analysis Department, Lomonosov Moscow State University (1-52, Leninskie Gory, Moscow, 119234 Russia). E-mail: chistyakov.ivan@yahoo.com. Scopus Author ID: 57212444724, <https://orcid.org/0000-0002-1029-3690>

Ivan V. Grishov, Postgraduate Student, Control Problems Department, Institute of Cybernetics, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow, 119454 Russia). E-mail: ivan.grishov@yandex.ru.

Alexey A. Nikulin, Researcher, KS Kadrovyi consulting (2-7, Bolshoi Starodanilovskiy per., Moscow, 115191 Russia). E-mail: nikulin.alexey@huawei.com.

Mikhail V. Pikhletsy, Cand. Sci. (Eng.), Principal Researcher, Huawei (1/7 Altuf'evskoe sh., Moscow, 121614 Russia). E-mail: pikhletsy.mikhail@huawei.com.

Ilya B. Gartseev, Cand. Sci. (Eng.), Associate Professor, Control Problems Department, Institute of Cybernetics, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow, 119454 Russia); Lead Engineer of Key Projects, Huawei (1/7 Altuf'evskoe sh., Moscow, 121614 Russia). E-mail: gartseev@mirea.ru. ResearcherID: Y-6501-2019, Scopus Author ID: 55973474600, <https://orcid.org/0000-0001-5866-3411>