Title

WeAllWalk: An Annotated Data Set of Inertial Sensor Time Series from Blind Walkers

Authors

Flores, G

Manduchi, R

Publication Date

2023-12-11

Peer reviewed

WeAllWalk: An Annotated Data Set of Inertial Sensor Time Series from Blind Walkers

GERMÁN H. FLORES, University of California, Santa Cruz, USA
ROBERTO MANDUCHI, University of California, Santa Cruz, USA

We introduce WeAllWalk, a data set of inertial sensor time series collected from blind and sighted walkers using a long cane or a guide dog. Ten blind volunteers (seven using a long cane, one using a guide dog, and two alternating use of a long cane and of a guide dog) as well as five sighted volunteers contributed the data collection. The participants walked through fairly long and complex indoor routes that included obstacles to be avoided and doors to be opened. Inertial data was recorded by two iPhone 6s carried by our participants in their pockets and carefully annotated. Ground truth heel strike times were measured by two small inertial sensor units clipped to the participants' shoes. We also present an in-depth comparative analysis of various step counting and turn detection algorithms as tested on WeAllWalk. This analysis reveals interesting differences between the achievable accuracy of step and turn detection across different communities of sighted and blind walkers.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile devices**; **Accessibility technologies**;

Additional Key Words and Phrases: Inertial sensing, wayfinding, step counting, turn detection

ACM Reference Format:
Germán H. Flores and Roberto Manduchi. 2017. WeAllWalk: An Annotated Data Set of Inertial Sensor Time Series from Blind Walkers. *ACM Trans. Access. Comput.* 9, 4, Article 1 (November 2017), 28 pages.
https://doi.org/0000001.0000001

1 INTRODUCTION

For someone who cannot see, tasks such as finding one's own location or figuring out how to reach a certain location in a building can be daunting, especially if this person is not familiar with the building layout or if he or she has poor orientation skills. Lacking access to visual landmarks, a blind traveler can quickly become disoriented; and if he or she at some point finds himself or herself being lost, tracing back their own steps can be equally challenging. For this reason, many blind individuals avoid visiting new places (office buildings, hospitals, schools) without a sighted guide who can show them around and lead them to the desired destination. Without the ability to travel independently, people in this community may miss opportunities for education, employment, leisure, socialization, and participation.

Personal navigation systems are designed to provide their users with spatial information and directions when traveling to new places. While outdoor navigation is to some extent solved by the use of GPS, GPS is not an option for indoor navigation, and various alternative technologies are

Authors' addresses: Germán H. Flores, University of California, Santa Cruz, 1156 High Street, Santa Cruz, CA, 95064, USA, ghflores@ucsc.edu; Roberto Manduchi, University of California, Santa Cruz, 1156 High Street, Santa Cruz, CA, 95064, USA, manduchi@soe.ucsc.edu.

1936-7228/2017/11-ART1 $15.00
https://doi.org/0000001.0000001

being explored. Of course, systems for indoor navigation are useful not only for blind travelers; anyone may need directional information at times. Indeed, there is increasing commercial interest in technology that may help one locate a shop in a mall, a room in a building, or one's own car in a parking lot. Several research groups have started building assistive applications on top of this technology, adapting it to the particular needs of specific communities of users.

This contribution focuses on systems that support indoor wayfinding using dead reckoning from inertial sensors. This approach has the advantage that it requires no external infrastructure (unlike iBeacons or similar technologies), or use of a camera (unlike image-based technologies). We note that, until wearable cameras are socially accepted and widely used, users of a camera-based localization system would need to take pictures of the environment with their cell phone, something that for a blind person may be difficult and possibly awkward in social settings. In contrast, inertial sensing can be conducted with a smartphone conveniently tucked in one's pocket.

Dead reckoning is a technique that uses data from inertial sensors (and from magnetic sensors, when the data they produce is reliable) to estimate the trajectory taken by the user. In theory, data from a tri-axial accelerometer could be doubly integrated to compute one's location. In practice, this is only possible with sensors attached to the walker's feet; by detecting when one's foot is resting on the ground, it is possible to perform a zero-velocity update, thus largely limiting errors due to drift [7]. When the sensors are worn elsewhere on one's body or garments, a safer strategy is to use them for step counting, and to indirectly recover one's position using an estimated stride length, as well as orientation information from the gyroscope. Note in passing that many blind individuals prefer receiving information about distances in steps, rather than in feet or in time [25]. Various versions of this approach have been used to track a person walking in a place with known geometry (obtained, for example, from a floor plan) [13, 26, 29]. Even when the geometry of the environment is not known, it is possible, in principle, to use dead reckoning (e.g., by means of step counting and robust turn detection) to help a person re-trace a path taken in a building.

Step counting and turn detection with a smartphone placed in one's clothing can be performed reliably if one walks with a steady gait and in mostly rectilinear paths. Blind individuals, however, often exhibit body motion patterns during gait that are markedly different than those of sighted people [22] (e.g., due to "scuttling" [14]). In addition, cane users, who are trained to execute the 2-point touch or constant sliding technique [8], swing their cane-holding arm left and right, resulting in additional upper body rotation. As already observed elsewhere [14], step counting may be difficult (or require specific parameter tuning) to work robustly with these individuals and for any smartphone placement. Likewise, blind individuals, especially when walking in large spaces, and unless they use a guide dog, do not always walk on straight paths with sharp and clearly detectable turns. Rather, they often veer involuntarily, and need to correct their path when they realize that they are getting close to a wall or an obstacle.

This paper introduces a new, openly accessible and annotated data set of inertial sensor time series collected from blind individuals walking through relatively long and complex paths in realistic conditions, and carrying two smartphones in different locations on their clothing. The primary purpose for creating this data set was to allow other researchers to benchmark their algorithms (step counting, turn detectors, or other) on a common ground. This follows the example of other similar data sets (described in Section 2.3), with the critical difference that our WeAllWalk data was obtained from blind walkers (in addition to sighted walkers), using either a long cane or a guide dog. As importantly, this data set does not just contain measurements from people walking on a straight line, as in previous collections [9, 34]. Instead, our participants walked on multiple paths with different levels of complexity, including turns at 45, 90, and 180 degrees, as well as through doors that needed to be opened. While walking, our participants occasionally veered off the straight path, got caught in wall openings, and collided with obstacles. These events (which

are faithfully recorded in the WeAllWalk data set) are to be expected when walking without sight. We carefully annotated our measurement time series, indicating the start and end time of each such event. In addition, we provide ground truth data in the form of heel strike times, measured by accessory inertial sensors clipped to the participants' shoes. We believe that this annotated data is representative of typical situations encountered by blind walkers, and that it should be very useful for anyone who wants to benchmark their dead reckoning algorithms in realistic scenarios.

We also present a comparative analysis of a number of step counting and turn detection algorithms on the WeAllWalk data. Specifically, we looked for differences in mean errors among different communities of walkers (cane users, dog users, and sighted walkers), different algorithms, and different phone placement locations on the walker's body. We also investigated whether algorithm parameters optimization on one community of walkers can transfer to other communities.

This article is organized as follows: After the related work, presented in the next section, we introduce the WeAllWalk data set in Section 3. We describe the sensor platform, the paths and their characteristics; we introduce the participants to this study, the procedures that were followed, and our criteria to annotate the data collected. In Section 4, we present a comparative evaluation of step counting and turn detection algorithms, as applied to the WeAllWalk data. Section 5 has the conclusions.

The WeAllWalk inertial sensor time series data set is available at **http://n2t.net/ark:/b7291/d1cc7g**. It is released under the terms of the Creative Commons Attribution license (CC-BY-4.0).

This article is an expanded version of a previous conference paper with the same title [16]. The conference version of the paper only contained a very early and succinct experiment of step counting and turn detection application (with only one algorithm and no comparative evaluation). In addition, two more blind walkers (P9 and P10) contributed to the WeAllWalk data set after publication of our earlier article.

2 RELATED WORK

2.1 Indoor Navigation via Inertial Sensing

There has been increasing interest over the past decade in personal navigation systems that support users in determining their location and in finding a path to a desired destination. While outdoor localization can be obtained, at least with an accuracy of a few meters, via GPS, this is not possible indoors, where the GPS signal becomes too weak for reliable positioning. Indoor navigation represents the "last frontier," with whole conferences devoted to this subject [1, 2]. A variety of techniques have been proposed for indoor localization [15], including radio-frequency triangulation [34], image-based recognition [27], Bluetooth beacons [3], visual markers placed in specific locations [11], and dead-reckoning using inertial sensors (see survey by Yang et al. [43]). The use of inertial sensors for blind indoor wayfinding has also been considered by several authors [10, 12, 14, 28, 33, 36, 38, 43].

2.2 Step Counting

Automatic step counting (e.g., for physical activity tracking) has received considerable attention by the research and industry world alike. Commercial pedometers use sensors that can be embedded in shoes (e.g., Adidas Micropacers), in a smartwatch, in a smartphone, and attached to one's ankles or belt. We refer the reader to [42] for a review of different sensing modalities for step counting and other physical activity monitoring. A variety of algorithms have been proposed for stride event detection from inertial sensor time series; an excellent review of some of the main algorithms is presented in [9]. Recent approaches to step detection include the use of recurrent neural networks [13]. Sensor placement has a role in the characteristics of the data collected. For example, ankle

or foot worn sensors usually provide more accurate step counting [18] than waist worn sensors. However, step counting accuracy does not seem to be greatly affected by the specific location of the sensor on other parts of the body [9, 20] (including on head-mounted displays [6]).

Whereas the vast majority of step counting algorithms have been developed for able-bodied ambulators, some authors have addressed the performances of these algorithms with sensors carried by people with some level of mobility impairment. For example, [32] evaluated different algorithms with ten mobility-impaired geriatric patients, while [44] designed and tested robust stride event detectors for users with Parkinson's disease. In both cases, participants carried an accelerometer on a belt around their waist. While none of the blind individuals who contributed to the WeAllWalk data set could be considered to have a mobility impairment, use of a long cane or a guide dog may result in a gait pattern that is quite different than for sighted walkers [21].

2.3 Similar Data Sets

We are aware of two existing openly accessible data sets with inertial time series collected from walkers carrying a smartphone. These data sets are briefly described below. Other similar data sets exist, but with different sensors and body placement (e.g., foot-mounted sensors [5], or video and inertial data from a hand-held device [35]), which are not directly relevant to our intended use case.

The Walk Detection and Step Counting on Unconstrained Smartphones dataset [9] consists of time annotated sensor traces (accelerometer, gyroscope, and magnetometer) obtained from 27 participants walking a route at three different walking paces, and carrying one or two smartphones placed in various positions while walking.

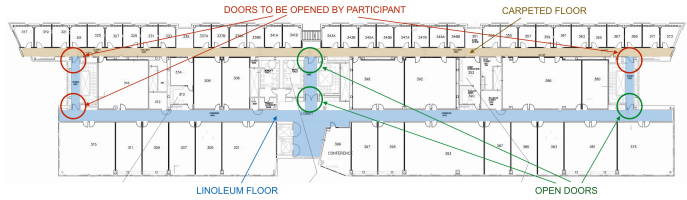
The OU-ISIR Gait Database [34] consists of walking data from 744 participants wearing four sensors (three units with accelerometer and gyroscope, and one smartphone containing an accelerometer) located in a belt around the participants' waist. Participants walked on straight paths at varying inclinations.

WeAllWalk differs from these prior data sets in two main aspects. First, it contains data from blind walkers, both using a long cane and a guide dog. Second, the paths traversed by our participants are much more complex and realistic than the straight routes considered in the previous data sets. The routes in WeAllWalk (see Figure 1 and 2) include turns at corridor junctions, active door openings, as well as sporadic stops or short re-routings due to involuntary collisions with objects or walls, as should be expected during regular blind ambulation. We carefully annotated the time series to identify intervals corresponding to walking in a straight line, taking a turn, or opening a door, as well as specific "features," such as when the walker stopped for a short moment, bumped into an obstacle or a wall, or deviated momentarily from the path, for example because he or she missed a door or got stuck in an opening in the wall.

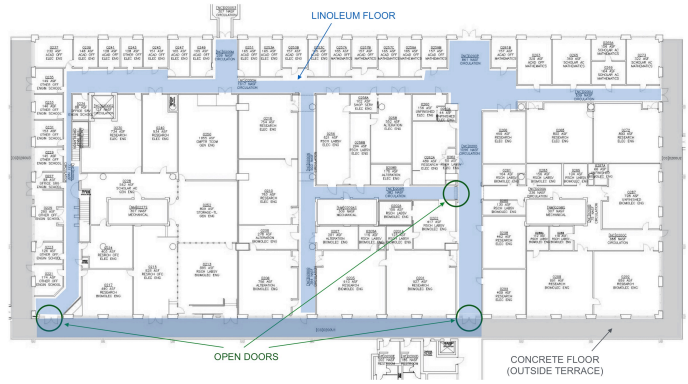
3 THE WEALLWALK DATA SET

3.1 Sensor Platform

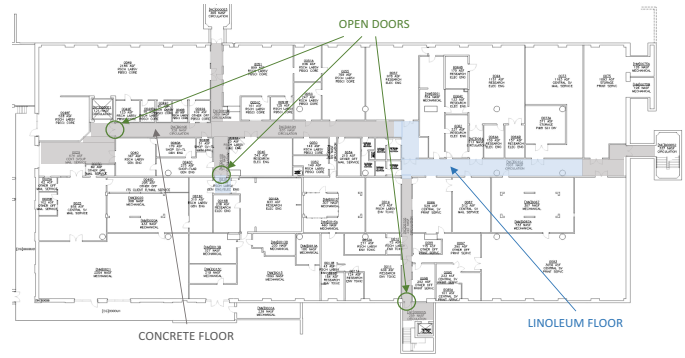
3.1.1 Sensors. Our participants carried two smartphones (Apple iPhone 6s), placed in different locations on their garments. Each smartphone recorded data from its tri-axial accelerometers, gyroscopes, and magnetometers. Data was sampled at a rate of 25 Hz. In addition, we recorded derived data produced by the iOS' Core Motion framework via Apple's proprietary sensor fusion algorithms. This derived data includes: the estimated direction of the gravity force; the device's actual acceleration (obtained by subtracting the estimated gravity acceleration from the data measured by the accelerometer); the corrected magnetic field; and the device's attitude (the 3-D rotation of the device with respect to a static reference frame). Each data sample was time-stamped with the clock of the phone that originated it.



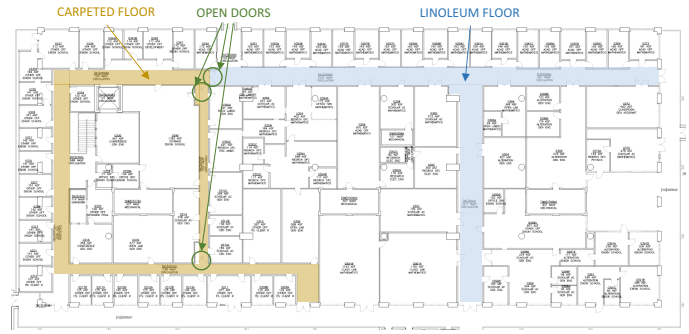
(a)



(b)



(c)



(d)

Fig. 1. The floor plans of the locations where data collection took place. (a): E2 3rd floor. (b) BE 2nd floor. (c) BE basement. (d) BE 3rd floor.

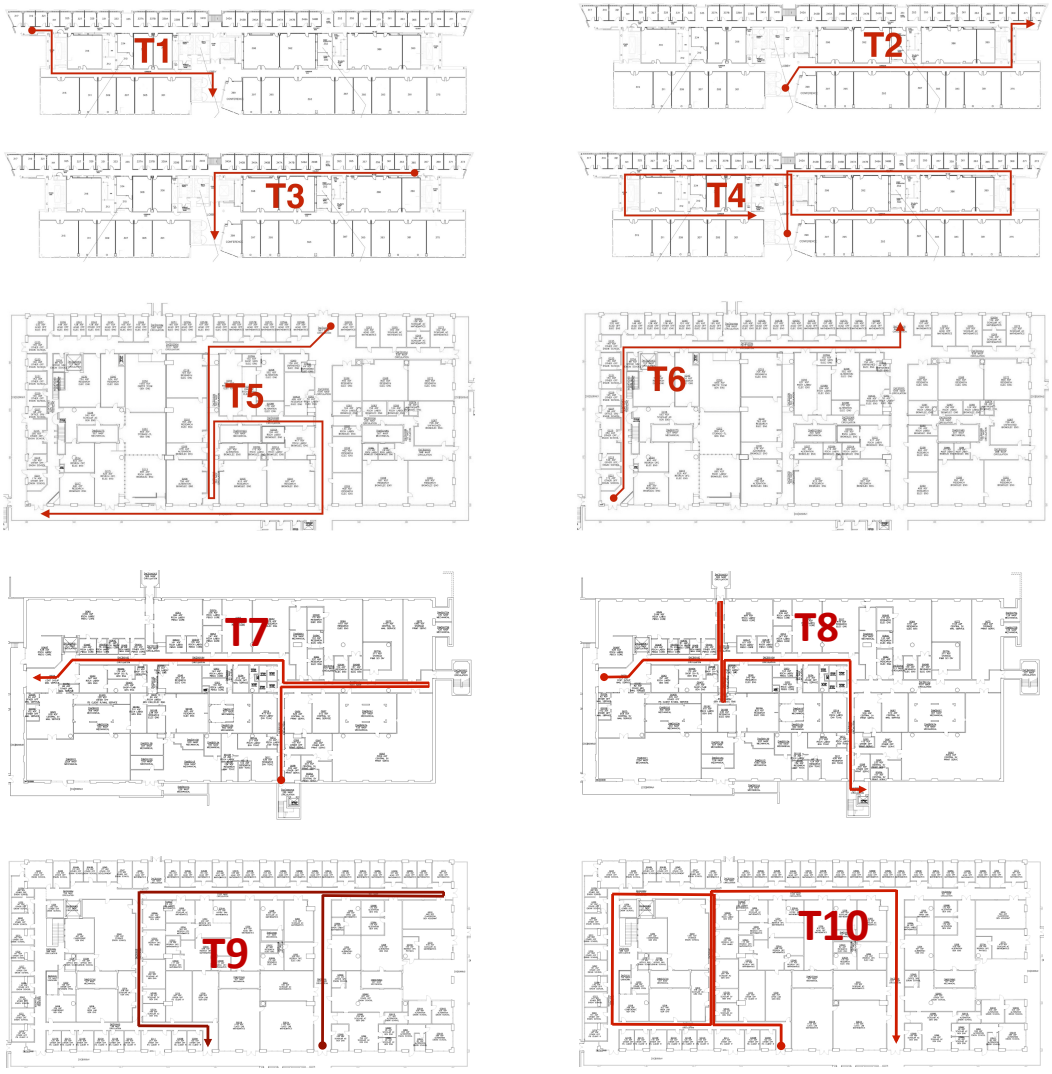


Fig. 2. The paths traversed by our participants. T1-T4: E2 3rd floor. T5-T6: BE 2nd floor. T7-T8: BE basement. T9-T10: BE 3rd floor.

In addition to the smartphones, our participants carried two small inertial sensor units clipped to their shoes (see Figure 3). We would like to emphasize that we do not assume or expect that blind walkers would wear these shoe-mounted sensors in their daily life. These sensors were added for the sole purpose of enabling ground truth step counting (since placement at the foot level enables very robust step detection [18]). This permits benchmarking of step counting algorithms from the smartphones' inertial sensors against the ground truth data from the foot-mounted sensors. We used MetaWear-CPRO[1] units (as shown in Figure 3), which contain a tri-axial accelerometer and

[1]https://mbientlab.com/product/metawearcpro

tri-axial gyroscope IMU from Bosch (BMI160). Both accelerometer and gyroscope produce data quantized at 16 bits within a programmable range of values. For the accelerometers, the available ranges are: $\pm 2g$, $\pm 4g$, $\pm 8g$, or $\pm 16g$ (where $g = 9.8m/s^2$). The gyroscopes, each of which measures angular rate around its own axis, can work with ranges of $\pm 125^\circ/s$, $\pm 250^\circ/s$, $\pm 500^\circ/s$, $\pm 1000^\circ/s$, or $\pm 2000^\circ/s$. For the experiments, we set the accelerometer range to $\pm 2g$, and the gyroscope range to $\pm 500^\circ/s$. The inertial sensor time series measured by the shoe-mounted sensors (sampled at 25 Hz) were recorded together with data from the smartphones, and later processed to detect the ground truth heel strike times. Foot strike events for each foot were detected from these sensors using data from the Y-axis gyroscope (as in [39]), processed by a modified version of the UPTIME algorithm [4] (see Figure 4).

Fig. 3. Examples of placement of the CPRO shoe-mounted sensor for ground truth step detection. The sensor is contained in the white small case, attached to a plastic padded clip.

All of the devices carried by our participants (two iPhone 6s, named *measurement phones*, and two foot-mounted sensors) were controlled via Bluetooth by a single iPhone 5 (called *control phone*) carried by one of the experimenters. The system made use of the Multipeer connectivity Framework[2] to communicate between multiple iOS devices, and of the MetaWear iOS Objective-C API to communicate with the MetaWear-CPRO sensors. The control phone was paired with both measurement phones. It broadcasted commands and received status updates (e.g., acknowledgement that a command was received, or battery life status from the MetaWear-CPRO sensors). The measurement phones were also paired with the MetaWear-CPRO sensors (one sensor per measuring phone). This pairing was activated remotely from the control phone. All of the sensor data from the MetaWear-CPRO sensors was streamed to the measurement phone paired to it, and recorded along with the measurement phone's own sensor data.

Once all Bluetooth pairings were executed, the control phone broadcasted a series of commands. These commands included: starting and stopping the data acquisition from the inertial sensors; synchronizing the measurement phones and MetaWear-CPRO sensors in order that all sensor readings reference the same starting time; saving all the sensor data from the smartphones and from the MetaWear-CPRO sensors; restarting the system for the next experiment. Importantly, remote interaction through the control phone removed the need for physical access to the measuring phones during the experiment. This means, for example, that our participants didn't have to take the measuring phones off their pocket and hand it to the experimenter between trials, in order to restart the system or save the data.

[2]https://developer.apple.com/documentation/multipeerconnectivity

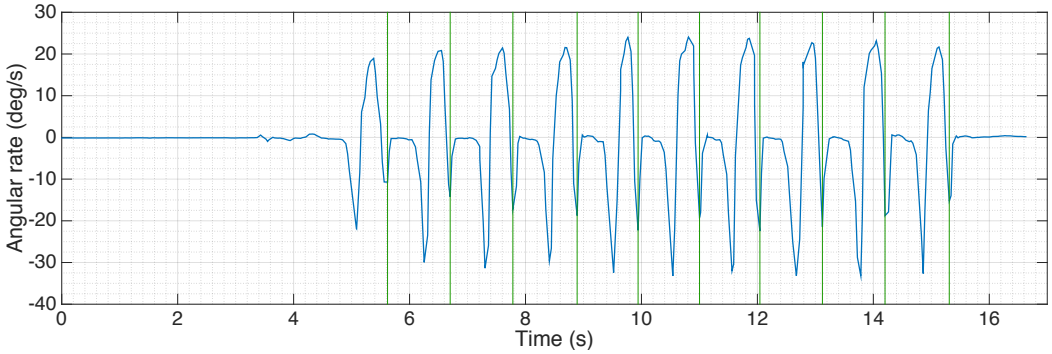


Fig. 4. Time series of measurements from the Y-axis gyroscope in the CPRO sensor clipped to one of our participants' right shoe during the calibration pre-trial. The green vertical lines represent the heel strike times as measured from this data.

3.1.2 Calibration Pre-Trial. As mentioned earlier, step detection from data from the shoe-mounted sensors is used as ground truth for subsequent processing of data from the measurement phones. It is thus imperative that the step counting algorithm used for the shoe-mounted sensors be carefully calibrated for each walker. For this purpose, we devised the following pre-trial calibration procedure.

Before starting to walk on the prescribed paths, each participant walked along a straight corridor for twenty steps. The time of each heel strike for each foot was recorded manually by a nearby experimenter, who tapped the screen of the control phone each time the participant placed a foot on the ground. (Note that manual step input was performed only in the pre-trial calibration phase.) This manually-input ground truth was then used to calibrate the parameters of the algorithm for step detection from data the shoe-mounted sensors (this calibration was performed off-line after data collection.)

3.2 Paths

Our participants walked along 10 different paths in two different buildings in our campus (called the E2 and the BE building in this paper). Note that the first version of the data set [16] contained only data from six paths, the first four (T1 to T4) located in the E2 building, and the last two (T5 and T6) located in the second floor of the BE building. In later work, we augmented the set with data collected by two more walkers, one walking along two trajectories (T7 and T8) in the basement of the BE building, and the other walking along two more paths (T9 and T10) on the third floor of the BE building.

Floor maps of the buildings are shown in Figure 1, while the paths' routes are shown in Figure 2. Paths were indoors and on level terrain (part of one path, T5, was on an outdoor terrace). We decided against including staircases in the paths, due to safety concerns. Routes where chosen to have a variety of lengths and complexities. The shortest path was about 75 meters long and only included one 90° turn; the longest path was about 300 meters long along an 8-shaped route, included seven turns and required the participant to open three doors. Four paths included one or more 180° turns, while five paths included one or more 45° turns. In order to minimize the effort required of our participants, the path order was designed in such a way that the end point of a path in the sequence corresponded to the start point of the next path.

For most of the time, participants walked in a corridor (with the width of the corridor varying from 120 cm to 210 cm). However, some paths included traversal of an open space (an elevator hall or an entrance hall) as well as of a passage next to a stairwell. In some cases, a turn was preceded or followed by a door that needed to be opened. In these cases, we informed the participant in advance of the presence of a door, and of whether the door had to be opened by pushing on a crowd bar, or pulled open by a handle. In two places, the path went through a door that required substantial force for opening; in these cases, one of the experimenters opened the door for the participant. The presence of doors is clearly indicated in Figure 1.

Floor surface or cover varied from industrial carpet to linoleum to rugged concrete (in the outside terrace). In addition, two industrial flat mats were placed in an elevator hall, and a metal plate was placed across a corridor. Some of our participants got their cane tip or their shoe briefly stuck at the edge of these floor coverings. Most environments were devoid of obstacles, although a few corridors contained large pillars, couches, chairs, tables, garbage bins, and obstacles in the form of appliances, which were kept on one side of the corridor. In these situations, we advised the participant to keep close to the opposite side of the corridor. Some corridors contained openings to rooms or to other corridors, and a few participants occasionally moved close to these openings and got caught in the wall corner; this typically caused a short stop before the participant was able to get back to the intended route. At times, participants also had to stop and move to the side to avoid walking into people who were standing in the corridor or were walking towards the participant. On the day participant P6 visited, some corridors were encumbered by one or more ladders due to ongoing work. In this case, we directed the participant by voice to avoid the ladder.

3.2.1 Path Segmentation and Annotation. The data set contains various levels of spatial annotations. Rectangles partitioning the space where data collection took place were defined on the floor maps, where rectangles represent either *straight segments* or *turn segments* (the latter connecting two straight segments at angles of 90°, 45°, or 180°). Each segment was assigned an ID. The horizontal coordinates of two opposing corners of each rectangle were recorded in a table. These coordinates were computed in terms of a reference system with axes oriented with the main orientations of the corridors, and origin placed at an arbitrary location (the top left corner of the floor map). An example of straight and turn segments for a path is shown in Figure 5(a). Each path can thus be defined by the ordered sequences of IDs of the segments composing the path.

3.3 Participants and Procedure

Ten blind volunteers and five sighted volunteers participated in the study. The blind participants were recruited through the network of acquaintances of the second author, while the sighted participants were graduate students or faculty members in our school. Note that the focus of this data set is on blind walkers; we added data from sighted participants only as a "control," for comparison in identical settings.

Nine of the ten blind participants (all except for P4) used a long cane while walking. Two of them (P1 and P8) repeated the trials using their guide dog. P4 only used her guide dog. Eight participants (P1-P8) walked through paths T1-T6; P9 walked through paths T7-T8; P10 walked through paths T9-10.

3.3.1 Blind Participants. Participant P1 was a 66-year-old woman who has been blind since she was very young. She had a guide dog, a Labrador Retriever, who was functioning, but close to retirement. P1 felt that her dog was becoming distracted and was not as good as he used to be at staying away from obstacles. For this reason, she had recently taken some orientation and mobility (O&M) classes in order to refresh her long cane skills. She walked paths T1 to T4 with the dog first, then again with the cane. She then walked paths T5 and T6, with the dog first, and then again with

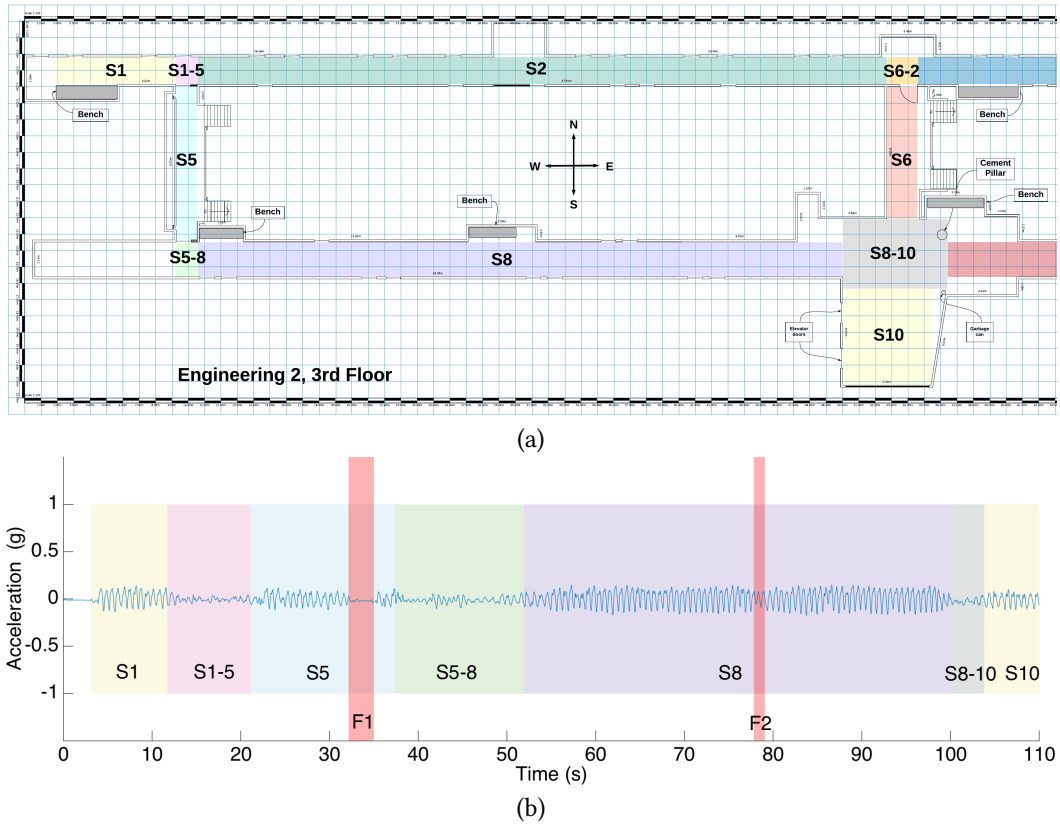


Fig. 5. (a): The walking area in E2 3rd floor, divided into segments (only half floor shown; see Figure 1). (b): Acceleration time series measured for P2 while walking along path T1 (see Figure 2), divided into segments traversed. Note the presence of two "features" (F1 and F2), which were recorded with their timestamps and associated annotations.

the cane. She felt that using the dog allowed her to walk on a straight line, while she tended to veer while walking with the cane; this was confirmed by our observations. Her dog, which she held on a harness with her left hand, often pushed her close to the right wall in the corridor. When walking with the cane, P1 sometimes got stuck in a wall opening and had to walk away from it to resume her path. She slid her pencil-tipped cane left and right, synchronized with her gait.

Participant P2 (aged 46) lost her sight over the past five years due to diabetic retinopathy (the diabetes also caused some neuropathy at her feet). She used a long cane for mobility, although she was looking forward to receiving a guide dog in the near future. She was still perfecting her mobility skills at the time of data collection, and felt that she was not moving as gracefully as other people in her condition. Her cane had a ball tip; she slid it left and right, synchronized with her gait. She often hit a sidewall with her cane, and sometimes bumped into obstacles along the way (e.g., a garbage bin or a chair).

Participant P3 was a 26-year-old woman who has been blind since birth. An expert cane user, she had a guide dog in the past. She admitted that her orientation skills are poor, so she was glad to hear that this study required no route memorization. P3 was able to walk on straight paths without much veering; however, she did get caught in a wall opening a few times. She used a cane with a

ball tip, sliding it on the floor in a swinging motion that, however, was generally not synchronized with her gait.

Participant P4 was a 65-year-old woman who lost her sight soon after birth. She didn't bring her cane, and thus was tested only with her dog, an energetic German Shepherd, who walked very fast as she held the harness with her left hand. The guide dog followed P4's commands faithfully, although at one point, while in a stairwell joining two corridors, the dog almost started leading P4 downstairs instead of walking straight past the staircase. P4 explained that the dog might have been wanting to walk to P4's husband, who was waiting downstairs in the parking lot.

Participant P5, aged 59, was a man who had lost his sight at 18 months of age. He never had a guide dog, and was not interested in one. He was an expert traveler, with excellent orientation skills. He often travelled independently by public transit. He had a peculiar way of using his pencil-tipped cane. Instead of swinging his cane left and right, he held it at an angle in front of him, and tapped it on the ground at regular intervals. He explained that, by listening to the sound and its echo, he could tell the presence of nearby surfaces. He walked, for the most part, with very little veering.

Participant P6 was a 68-year-old man. He had lost his sight due to a traumatic brain injury as a teenager. P6 used a telescopic cane with a round metallic glide tip, which he maneuvered in a swinging motion synchronized with his gait. He slid the cane on the floor except for the outside terrace with rugged concrete surface, where he instead tapped it (2-point touch [8]). P6 explained to us that he would normally use a different, heavier cane when walking outdoors. He was able to walk in straight lines and avoided almost all obstacles, without hitting any wall or being caught in wall openings.

Participant P7 was a man, aged 46, who had been blind since birth. He had excellent orientation skills and regularly travelled even long distances using public transportation. He used a single piece long cane with round metallic glide tip, which he slid on the floor in a swinging movement synchronized with his gait. In our trials, he walked with little veering. In a couple of occasions, he bumped his shoulder onto large obstacles along the path.

Participant P8 was a 69-year-old woman who lost her sight progressively during her young age. Similar to P1, she walked all paths twice, one time with her guide dog and the other time using a long cane (pencil tip). She was a proficient traveler, yet she often times veered off the straight direction when walking along a corridor and had to correct her path.

Participant P9, aged 22, was the youngest participant in our study. An expert traveler, he walked fast using the two-point touch technique, and was able to maintain a straight direction in most cases.

Participant P10, aged 62, lost his sight as a young man. He was a proficient cane user, and occasionally employed echo-location, producing clicking sounds with his mouth.

3.3.2 Smartphone Placement. Each participant was asked to choose a comfortable location for the two measurement smartphones. Preferences varied: some participants chose to place the smartphones in a front or back pants pocket, while other opted for a holster clipped to their belt or a jacket pocket at waist or breast height, or they kept a phone tucked under their shirt at shoulder level. Table 1 shows the different phone placements for all of our participants. Informal surveys (e.g., [23]) have shown that the majority of people keep their phone in their pocket, and for this reason we didn't consider placement of the smartphones in the participants' handbag or backpack. In addition, step counting with smartphones kept in a handbag was shown to be inaccurate [9] due to extra swinging of the bag. For the same reason, our participants generally kept their phone tightly close to their body. We also didn't consider the case of a smartphone held in one's hand while walking [37], as this may be inconvenient for blind people who already have one hand occupied holding a cane or a guide dog.

Table 1. The list of participants, with the location of the measurement phones in their garments. Phone locations are color coded to represent the categories discussed in Section 4.1.3.3 (Pants pocket-front, Pants pocket-back, Jacket pocket, Chest level).

	Community	iPhone1 Placement	iPhone2 Placement
P1	Blind:Cane	Left breast pocket	Jacket right side pocket
P2	Blind:Cane	Jacket left side pocket	Tucked under shirt on right shoulder
P3	Blind:Cane	Pants left front pocket	Pants right back pocket
P5	Blind:Cane	Pants left front pocket	Pants right back pocket
P6	Blind:Cane	Holster clipped to front right belt	Pants left front pocket
P7	Blind:Cane	Jacket right side pocket	Pants left front pocket
P8	Blind:Cane	Pants right front pocket	Pants left front pocket
P9	Blind:Cane	Pants right front pocket	Pants left front pocket
P10	Blind:Cane	Shirt left front pocket	Pants right front pocket
P1	Blind:Dog	Left breast pocket	Jacket right side pocket
P4	Blind:Dog	Jacket left side pocket	Pants right back pocket
P8	Blind:Dog	Pants right front pocket	Pants left front pocket
P11	Sighted	Shirt left front pocket	Pants left front pocket
P12	Sighted	Shirt left front pocket	Pants right back pocket
P13	Sighted	Jacket right side pocket	Pants left front pocket
P14	Sighted	Pants right front pocket	Pants left front pocket
P15	Sighted	Pants left back pocket	Pants right front pocket

3.3.3 Procedure. After signing the IRB-approved consent form, each participant was shown the CPRO sensors in their clip cases, and asked to clip each sensor case to the back, if possible, or to the side of their shoe (see Figure 3). (Note that participants were advised in advance of their visit to wear comfortable shoes, and to wear clothing with pockets.) Then, participants were asked to choose a location on their garments for their two measurement smartphones, as discussed in Section 3.3.2. Participants were advised not to pay attention to any speech produced by the smartphones (which were programmed to utter short synthetic speech verification sentences upon successful pairing with the control phone). Participants were also advised to begin walking only when prompted by an experimenter, and to walk straight until asked by the experimenter to turn left or right (or, in the case of path T5 or T7-T9, to turn around); to push or pull open a door; or to stop at the end of the path. These were the only verbal directions provided to the participants, except for occasional safety warnings (e.g., as mentioned earlier, participants were advised to walk closer to one side of a corridor if there were obstacles on the other side).

No training on the use of the system was necessary, since the task was for the participants to simply walk naturally. Each participant first went through the pre-trial described in Section 3.1.2 for ground truth calibration. Then, he or she was accompanied to the start position of the first path, and asked to start walking in the designated direction. Before the start of each path, participants were oriented to face the correct direction; this was particularly important for paths T2, T5 and T6, which started with diagonal traversal of an entrance or elevator hall. All trials with blind participants were supervised by two experimenters. One of the experimenters managed the start and end of data collection from all sensor platforms via the control phone, and recorded videos of all sessions by means of a GoPro HERO Session camera attached to a head strap. The other

experimenter walked at a close distance behind or sometimes in front of the participant, and was in charge of ensuring the participant's safety.

It is conceivable, although unlikely, that participants P1 and P8, who walked all routes in each building with the guide dog first, then with the long cane, might have memorized one or more routes during the first traversal. However, this would likely have no effect on the collected data, given that in any case the participants followed directions from the experimenter.

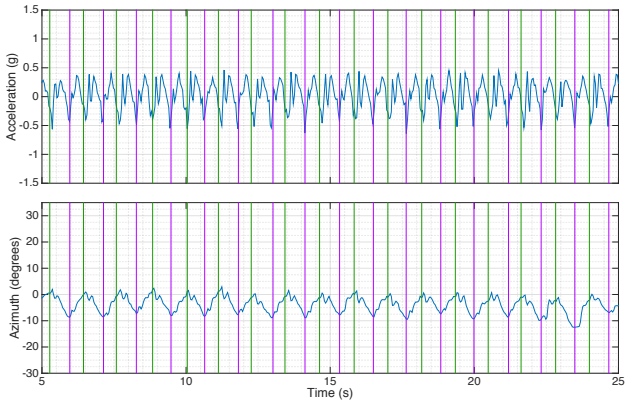
Figure 6 shows an example of time series collected during a straight path in route T3 for three individuals: a sighted participant, a blind participant using the long cane, and a blind participant using a guide dog. The first subfigures plot the projection of the tri-axial accelerometer measurements onto the principal component of its time series (that is, onto the direction where this vector has its largest variance). The azimuth represents the angle of the phone around the vertical (gravity) direction, referenced to an arbitrary horizontal axis. This data was obtained from the iOS CoreMotion Framework. (Note that the magnetometer was not used for azimuth computation, as we found that it decreases the quality of the azimuth in indoor environments.) The plots also display the heel strikes times (shown by vertical lines) for each foot. Observation of the azimuth time series in these figures provides some insight into the gait characteristics of each individual. In particular, one can notice how the sighted walker maintained a steady heading direction (with oscillation due to natural body swinging), whereas the azimuth time series of the blind walker with a cane shows a more variable pattern, with variation in heading direction as large as 20°. The blind participant using a guide dog maintained a more stable heading direction, but with a wider swinging action.

3.4 Data Annotation

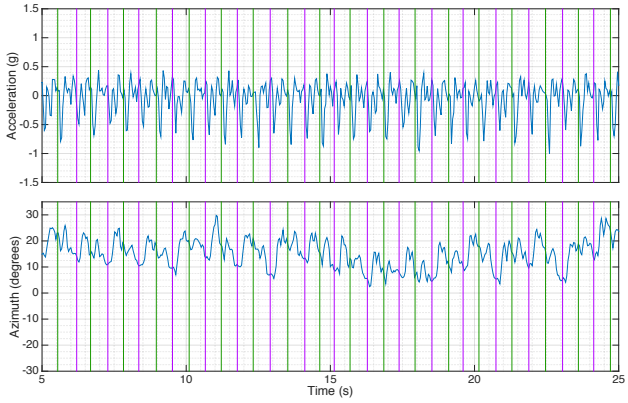
After completion of all trials for a participant, the data from all sensors was offloaded to a desktop computer for post-processing. In particular, all data streams were synchronized as discussed in Section 3.1.1. The video streams collected from the GoPro camera were synchronized to the same time base used for the sensors. The heel strikes times for each foot (computed by processing the shoe-mounted sensors) were also recorded.

The time lapse between the beginning and end of each route traversal was divided (by visual inspection of the video) into contiguous intervals, where each interval corresponded to the participant either walking along a straight segment in the path, or taking a turn. A *turn interval* lasts from the time the walker started the turn (which at times involved opening a door), till the time the walker began walking straight after turning. A *straight interval* joins two turn intervals. In general, the location of a walker at the beginning and at the end of a turn interval closely matches the edges of the associated *turn segments* in the path, as defined in the floor map (Section 3.2.1). Hence, from the recorded time intervals, it is possible to ascertain the times at which the walker crossed the edge between contiguous segments in the map. The direction of traversal of each segment is recorded in the annotation file, together with the start and end time of each interval, and with the number of steps taken during the interval. In addition, we recorded each *turn time*, along with the associated *turn amount* (±45°, ±90°, or ±180°). It should be clear that the definition of a turn time is somewhat arbitrary: rather than an instantaneous event, a turn is executed over an interval of time. We simply selected (based on analysis of the videos) a point in time approximately in the middle of such an interval.

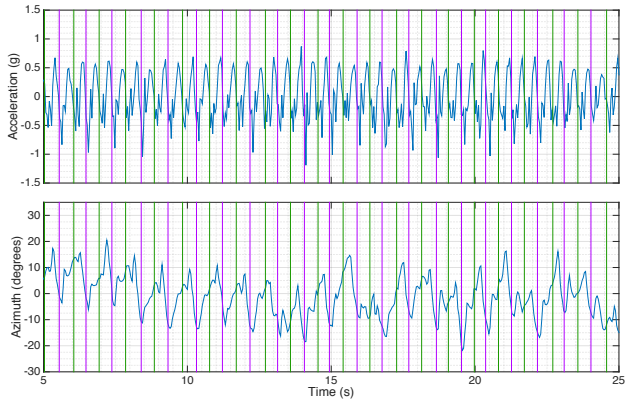
As an example, traversal of route T1 (defined in Figure 2) for one of our participants was divided into the seven contiguous time intervals shown in Figure 5(b). During each time interval, the walker found herself within the corresponding straight or turn segments as recorded in the map (Figure 5(a)).



(a) A sighted participant



(b) Participant P3 using a long cane



(c) Participant P6 using a guide dog

Fig. 6. Sample time series of measured accelerometer and azimuth data from different participants. The magenta and green vertical lines mark the left and right heel strikes.

In addition to the segmentation into straight and turn intervals, we created annotations of particular events such as opening a door, bumping into an obstacle, being caught in a door opening, or stopping momentarily (see Figure 7). These events are normally associated with anomalous characteristics in otherwise regular inertial data time series (see Figure 8 and 9). For example, in Figure 5(b), features are marked as elongated red rectangles. In this case, the first feature refers to the participant stopping towards the end of segment S5 as she listened to verbal directions given by the researcher. The second feature (segment S8) refers to the time when she hit a corner of a side wall with her left arm.

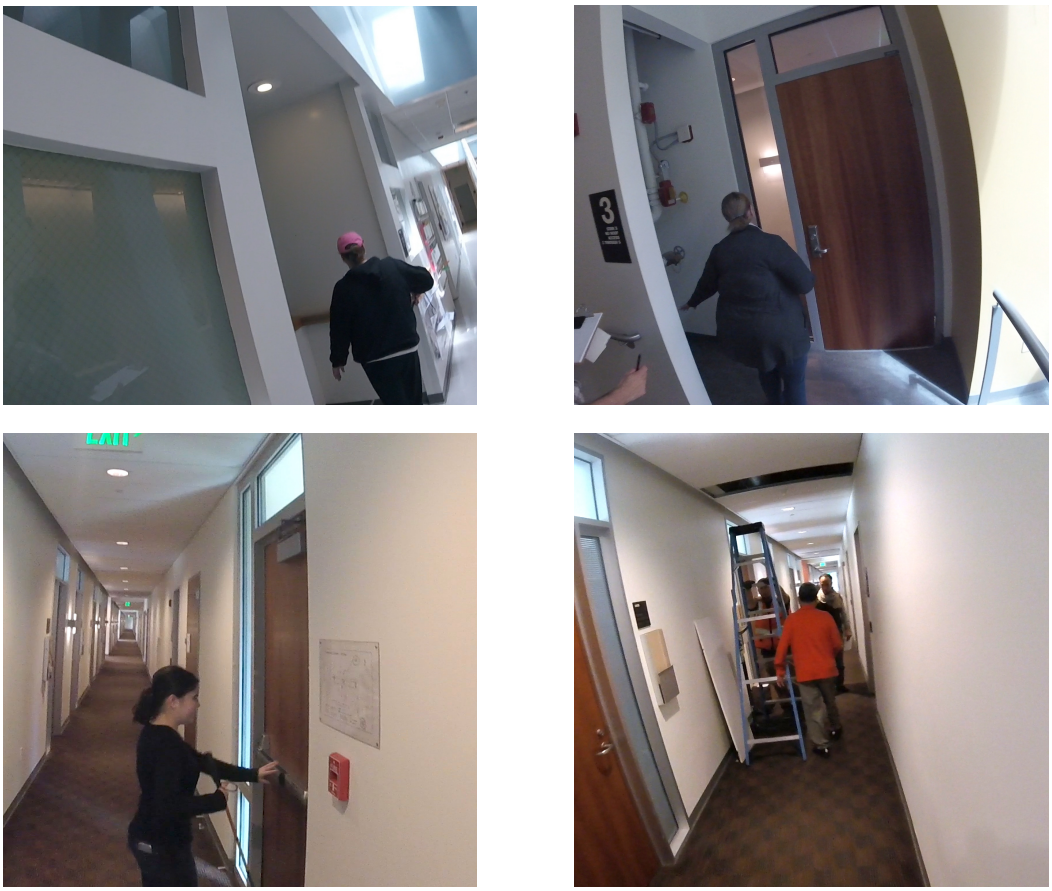


Fig. 7. Four of our blind participants dealing with specific situations. Top two images: being caught in a wall opening. Bottom left: pushing open a door. Bottom right: avoiding an obstacle (a ladder) in the way.

We noted that when participants were engaged in tasks such as opening a door, the shoe-mounted sensors sometimes detected "phantom steps" when in fact the participants were simply balancing themselves on their feet. We did not manually remove these phantom steps, as they occurred only sporadically in our study. The annotation file, which is stored using the Extensible Markup Language (XML) format, also includes other relevant information such as the type of mobility tool used, as well as some general gait pattern observations.

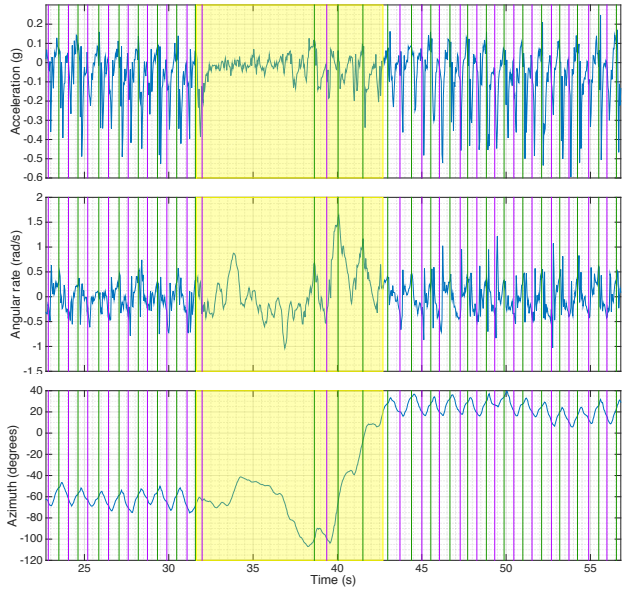


Fig. 8. Sensor data from a participant pulling a door open then making a left turn.

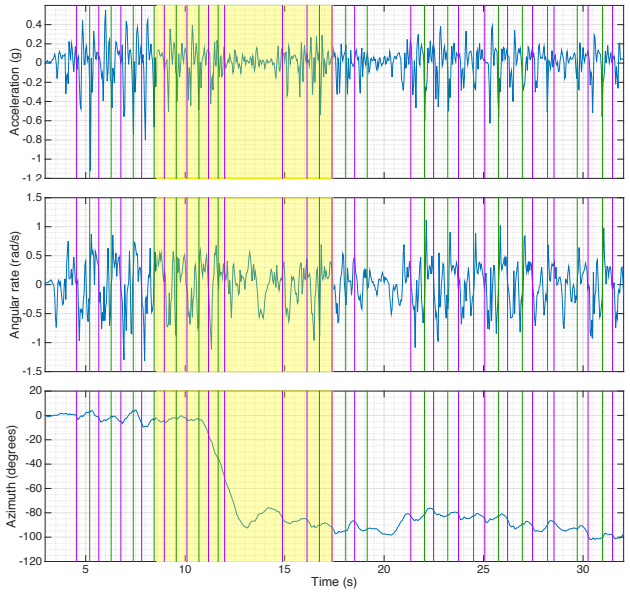


Fig. 9. Sensor data from a participant making a right turn then pushing a door open.

4 STEP COUNTING AND TURN DETECTION BENCHMARKING

In this section, we present a comparative analysis of two different classes of algorithms (step counting and turn detection) as applied to the WeAllWalk data set. Step counting and turn detection take on particular importance as critical components of virtually any dead reckoning localization

system for pedestrians in indoor environments [13, 26, 29]. WeAllWalk enables benchmarking of new or existing algorithms for this type of applications on a variety of walkers in different situations.

Of particular interest for our analysis is the difference in performance of these algorithms when measured across the different "communities" of walkers, in our case: blind walkers using a cane (*Blind:Cane*); blind walkers using a dog (*Blind:Dog*); and sighted walkers (*Sighted*). We also looked at differences between the considered algorithms using appropriate metrics, as well as at the potential influence of the phone location on one's garment on the performance of a particular algorithm.

Finally, we asked whether optimizing an algorithm's parameter on a certain community of walkers makes it less effective when used on another community, as discussed in detail in the following. We addressed this question by considering two alternative modalities for training data set design, where we adhered to the standard norm of "training" an algorithm on a different data set than the one used for benchmarking. Our training modalities are:

Train on sighted: Data from all sighted walkers was used to train the algorithms, which were then benchmarked on data from all blind walkers. This approach is justified by the consideration that data from sighted walkers is, in general, more easily available than data from blind walkers (e.g., from existing data sets collected from sighted walkers). Thus, one may ask how an algorithm trained on sighted walkers would fare when used on data from blind walkers.

Stratified leave-one-out: In this case, the three communities were considered independently. For each community, algorithms were benchmarked for each walker using parameters trained from all other walkers in the same community. Note that this is a standard cross-validation technique (*leave one out*), one that is particularly suitable for small size samples [19].

Remember that participants P1 and P8 used both cane and guide dog (in separate sessions, see Table 1). In our statistical analyses, both P1 and P8 were assigned two different levels of the factor *Participant*: one when using the cane, and the other when using the guide dog. In other words, we considered data from sessions with P1 or P8 using the cane as being collected from a different individual than when using the guide dog. This procedure is justified by the fact the same participant walked in a noticeably different way when holding a long cane than when handling a guide dog.

4.1 Step Counting

We conducted initial tests with several of the techniques mentioned in Section 2.2, as well as with some variations thereof, and selected a total of six algorithms to experiment with. Some of these algorithms operate on the accelerometer data, while others use data from the gyros. All of the algorithms have some parameters that must be learned from training data. Importantly, we only train and test the step detection algorithms on data acquired while traversing straight segments in the paths, where gait is assumed to be regular, and avoided segments labeled as features. This is because the notion of "step" is not well defined in such situations (e.g., when someone bumps into an obstacle or needs to open a door). In general, we argue that step counting only really matters during regular ambulation, as an indirect way to measure distances traversed.

For most of the algorithms, parameter optimization is performed via exhaustive search in a discretized parameter space. The Matlab implementation of all algorithms are available online and included with the WeAllWalk data set.

4.1.1 Step Counting Algorithms. The six algorithms considered in our experiments are briefly described below, together with a list of parameters to be tuned for each algorithm.

WPD [9]: The Window Peak Detection (WPD) algorithm runs a moving average window on the smoothed accelerometer magnitude to find peaks associated with a heel strike (where the accelerometer magnitude is computed from all three accelerometer axes). Parameters:

- Window size ($MovAvr_{win}$)
- Peak threshold ($Peak_{win}$)

AMPD [41]: Automatic multiscale-based peak detection (AMPD) is a generic algorithm for peak detection in a signal. As in the case of WPD, it is used here to find peaks in the accelerometer magnitude (under the assumption that these peaks are associated with heel strikes) by detecting local maxima. The Savitzky-Golay (S-G) filter [40] is used to smooth the accelerometer magnitude before computing the local maxima. Parameters:

- S-G polynomial order (P_{coeff})
- S-G sample frame length (fl_{sample})

UPTIME [4]: UPTIME (Ubiquitous Pedestrian Tracking usIng Mobile phonEs) uses the de-trended magnitude of acceleration in a finite state machine (FSM), with six states to identify the peaks associated with heel strikes. Parameters:

- FSM thresholds (Thr, Pos_Peak_Thr, Neg_Peak_Thr, Neg_Thr)

HMM-acc [9, 30]: This algorithm trains a Hidden Markov Model (HMM) to discern the different phases during a gait period. A model with two hidden states is used to locate and detect the different gait movement phases (heel strike, heel off) captured by the de-trended magnitude of the accelerometer data. Note that the original algorithm [9, 30], which was designed for foot-mounted sensors, used data from the gyro aligned with the X axis. In our experiments, this data produced very poor results as compared with de-trended accelerometer data. Parameters:

- 2x2 transition probability matrix (A)
- Emission probability parameters (μ, σ)
- Prior probabilities (π)

ZC-gyro [24]: This technique searches for zero crossings (ZC) within a moving window of the data from the gyro aligned with medial-lateral axis. Assuming that the smartphone was kept approximately vertical and with its face parallel to the walker's body, this axis is approximately aligned with the phone's X axis. This was not the case for one of the phones carried by participant P6, which was held horizontally in a holster clipped to his belt. In this case, data from the gyro in the Y-axis was used instead. Zero-crossings are only computed between positive and negative peaks larger than a threshold. A discrete Butterworth low pass filter (BW) was applied to smooth the signal before zero-crossing detection.

- BW cut off frequency (f_{coff})
- BW filter order ($Order_{coeff}$)
- Moving Window size (Mov_{win})
- Peak threshold ($Peak_{thr}$)

ZC-acc: This is a variation of the ZC-gyro algorithm, that uses de-trended accelerometer magnitude instead of gyroscopic data. (We used the *detrend* function from Matlab, which removes the best straight line fit to the time series.) Its parameters are the same as for ZC-gyro.

4.1.2 Error Metrics. The quality of the considered step counting algorithms was measured using two different metrics. The first metric (*SC-Error 1*) looks at the number of steps detected within each time interval $[T_{i-1}, T_i]$ separating two consecutive ground-truth heel strikes. Ideally, exactly one step should be detected within $[T_{i-1}, T_i]$. If no steps are detected in $[T_{i-1}, T_i]$, then an *undercount* event is recorded. If $n > 1$ steps are detected within that interval, then $n - 1$ *overcount* events are

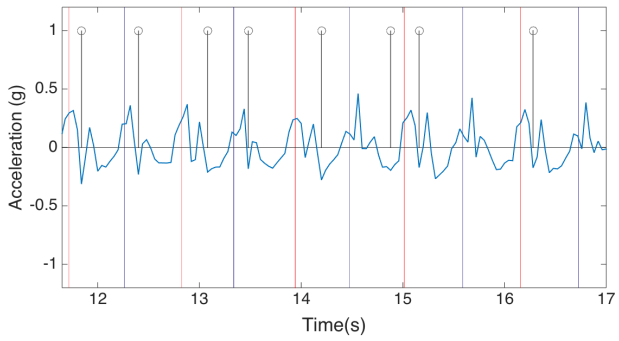


Fig. 10. De-trended accelerometer data with step detected by the UPTIME algorithm (black lines with circles) as compared to ground truth steps (red line). Note that one step is missed by the algorithm, resulting in one undercount event.

recorded (see Figure 10) The cumulative number of undercount and overcount events are computed and normalized (divided) by the number of ground-truth steps.

The second metric (*SC-Error 2*) simply computes the difference between the number of detected steps within each segment and the number of ground-truth steps within the same segment. The difference between the two is recorded as an undercount value if negative, as an overcount if positive. Undercount and overcount values are then summed together over all segments, and normalized by the total number of ground-truth steps.

Note that the normalized undercount and overcount values obtained by the SC-Error 1 metric is always larger than or equal to the corresponding values computed by the SC-Error 2 metric, as missed counts or double counts between a time interval $[T_{i-1}, T_i]$ may even out over multiple steps. While the more lenient SC-Error 2 metric may be appropriate when measuring step counts over long hauls, the more conservative SC-Error 1 metric may be useful when fine-grained tracking is desired.

When training and benchmarking the algorithms, the sum of overcount and undercount rates is used for both error metrics.

4.1.3 Step Counting Results. We processed the WeAllWalk data to benchmark the different step counting algorithms for the different communities represented in the data set. Figure 11 and 12 show the average SC-Error 1 and 2 for each community and for each considered algorithm under the Stratified Leave-One-Out training modality. All tests were conducted by taking the logarithm of SC-Error1 or 2 as dependent variable, as this was shown to increase Gaussianity of the residuals of linear fit, based on visual inspection of the Q-Q plot. In the case of SC-Error1, the residuals were actually confirmed Gaussian by the Shapiro-Wilk normality test.

4.1.3.1 Effect of Training. We evaluated whether there was any difference between the two training modalities considered (*Train on Sighted* and *Stratified Leave-One-Out*). Note that error data is only computed on blind walkers for this test.

A repeated measures ANOVA with blocking factors of *Participant*, *Algorithm*, and *Phone placement* was performed, and we did not observe any statistically significant difference in the mean value of either SC-Error1 or 2. In light of this result, and for simplicity's sake, we decided to only consider the *Stratified Leave-One-Out* modality in the analyses discussed next.

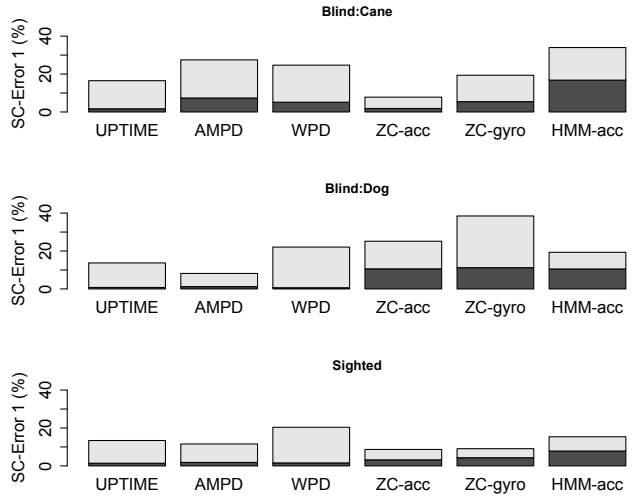


Fig. 11. Values of SC-Error 1 averaged over participants for the different communities and different algorithms considered for the Stratified Leave-One-Out training modality. Gray bars indicate undercount rates, while black bars indicate overcount rates.

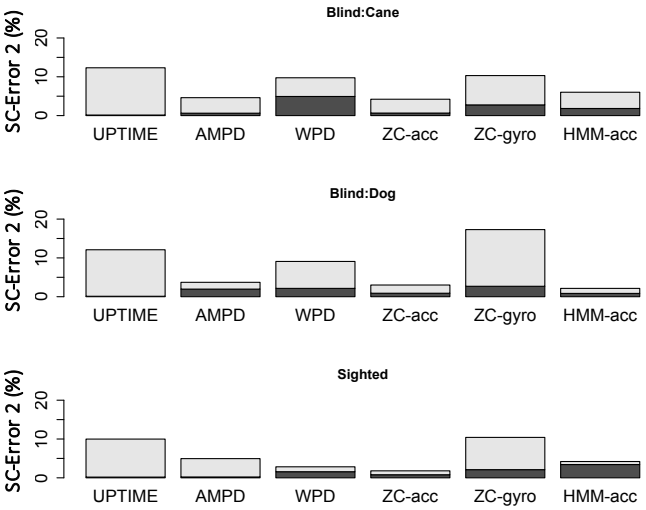


Fig. 12. Values of SC-Error 2 averaged over participants for the different communities and different algorithms considered for the Stratified Leave-One-Out training modality. Gray bars indicate undercount rates, while black bars indicate overcount rates.

4.1.3.2 Effect of Algorithm and Community. This analysis addressed the effect of different step counting algorithms and of different communities of users on the measured errors. The goal was to ascertain whether some algorithms are significantly better than others, and whether step counting produce significantly different errors for walkers in different communities. We tested the null hypothesis of equal error means for different levels of the factors *Algorithm* and *Community*, with blocking factors of *Participant* and *Phone placement*. Main effects and interactions were discovered

using ANOVA at significance level $\alpha = 0.05$. Post-hoc multiple comparisons were conducted using Tukey's range test. The results are discussed below for the two error metrics considered.

SC-Error 1: A significant effect of both Algorithm (p=4e-7) and Community (p=3e-3), as well as of their interaction (p=4e-5), was found. Post-hoc analysis of main effects follows.

Community: The mean error for walkers in the Sighted community (13.1%) was found to be significantly smaller than for both the Blind:Cane community (21.6%) and the Blind:Dog community (21.1%).

Algorithm: The mean error of WPD (22.7%) was found to be significantly larger than AMPD (18.3%), UPTIME (14.9%) and ZC-acc (11.6%, lowest). ZC-acc was also found to have mean error significantly smaller than HMM-acc (24.8%).

Since significant interaction was found, we studied the marginal effects of *Algorithm* at the different levels of *Communities* (significance of effect was found at all three levels).

Community=Blind:Cane: The mean error for ZC-acc (7.8%, lowest) was significantly lower than for WPD (24.7%), AMPD (27.5%) and HMM-acc (34.0%). HMM-acc. had a significantly higher mean error than UPTIME (16.5%). We also noted that, within this community, *Participant* had a significant effect in SC-Error 1 (p=1e-2), with values ranging from 13% for P3 to 26% for P6. However, no significant difference in mean was found in post-hoc pairwise comparison.

Community=Blind:Dog: Both AMPD (8.2%, lowest) and UPTIME (13.7%) had significantly lower mean error than ZC-gyro (38.5%).

Community=Sighted: WPD (20.4%) had significantly higher mean error than both ZC-acc (8.7%, lowest) and ZC-gyro (9.1%).

SC-Error 2: A significant effect of both *Algorithm* (p=4e-11) and *Community* (p=3e-2), but not of their interaction, was found. Post-hoc analysis of main effects follows.

Community: A significant difference in mean error was found between *Blind:Cane* (7.9%) and *Sighted* (5.7%).

Algorithm: The mean error of ZC-acc (3.2%, lowest) was found to be significantly lower than for ZC-gyro (11.7%) and UPTIME (11.5%). UPTIME also had higher mean error than both AMPD (4.5%) and HMM-acc (4.6%). ZC-gyro was found to have higher mean error than WPD (7.3%) and AMPD (4.5%).

4.1.3.3 Effect of Phone Placement. As explained in Section 3.3.2, our participants chose where to place the two measurement phones during the trials (see Table 1). In order to study the effect of phone placement on the mean step counting error, we first clustered the various phone locations into four groups:

- *Pants pocket - front* (both left and right pocket)
- *Pants pocket - back* (both left and right pocket)
- *Jacket pocket* (at waist level)
- *Chest level* (this includes a phone placed in breast pocket, or tucked at level shoulder under the shirt)

The single case of a phone kept in a holster clipped to the belt (P6) was removed from this analysis.

Repeated measures ANOVA with blocking factors of *Participant* and *Algorithm* found no significant difference in the mean value of either SC-Error1 or 2 for the different phone locations.

4.1.4 Discussion. Several interesting observations can be drawn from our investigation of step counting algorithms as applied to data in WeAllWalk. Our expectation that people in the different communities considered may have different walking characteristics was indirectly confirmed by the fact that the same step counting algorithms perform differently (in terms of mean error) across

these communities. Our data found a significantly larger step counting error for blind walkers using a cane or (for SC-Error 1 only) a guide dog, than for sighted walkers. In particular, one should expect that for walkers using a cane, a step counting algorithm may produce an error that is approximately 50% higher than for a sighted person. Speculating on the exact reason for this difference is beyond the scope of this article.

Among the algorithms considered, no clear winner was found, although our analysis does provide statistical evidence in favor of some techniques. For example, for blind walkers using a long cane, the ZC-acc algorithm, with a low SC-Error 1 value (7.8%), compared significantly favorably against three other competitors. The same ZC-acc algorithm produced significantly lower mean SC-Error 2 values than two other techniques when tested across all communities of walkers.

Our analysis did not find a significant dependence of step counting performance on the location where the iPhone is kept, thus confirming earlier results [6, 9, 20].

Interestingly, the performance of the considered step counting algorithms does not seem to be affected by the specific community (blind or sighted) the algorithms are optimized for. This seems to indicate that optimizing a step counting algorithm using data from a training sample of sighted walkers may be acceptable even when the algorithm is to be applied to data from blind walkers.

4.2 Turn Detection

An indoor route can often be expressed in terms of a sequence of turns, along with the number of steps taken in the path between two consecutive turns. For example, one may specify a route as: "Walk straight through this corridor for about 50 steps, then make a left at the junction, walk for 20 more steps and take a right at the first corridor." Note that for most buildings, corridors intersect at angles of 90° or, in some cases, 45°. Robust detection of turns from inertial data measured by a smartphone, combined with step counts between turns, may help travelers keep track of their progress in an indoor route. If a walker at some point feels lost, he or she may be able to return to the starting point (e.g., an entrance door) by simply following the sequence of recorded turns in reverse order.

4.2.1 Turn Detection Algorithms. A turn while walking determines a change of the walker's heading direction, and as such it can be immediately noted in the time series of azimuth data. The azimuth (heading direction) angle can be obtained via fusion of data from the phone's gyroscope and accelerometer, and possibly magnetometer. Specifically, we read azimuth from CMAttitude object in the iOS Device Motion framework, using a fixed reference frame with the Z axis pointing downwards.

Here is a very simple 90° turn detection algorithm based on azimuth data. First, divide the 360° angular span into sectors $\{k \cdot 90^\circ + \alpha_0\}$ for integer $0 \leq k \leq 3$, where α_0 is an appropriate bias accounting for the angle between the X axis of the reference system and one of the main orientations of the corridors in the building. Then, determine the index of the sector containing the current azimuth angle. If this index has changed with respect to the prior measurement, a turn event is declared. For example, assuming that $\alpha_0 = 10^\circ$, a turn would be detected if the previous azimuth measurement was 185° and the current measurement is 192°. This naïve algorithm, however, may fail in the case of wide swinging or veering during walking, which may trigger false turn detection (Figure 13). The two algorithms described below, originally proposed by the authors in [17], can accommodate for temporary swinging or veering by describing the time sequence of heading measurements using a suitable dynamic model. Note that, while in this work we only consider turns by ±90° or 180°, both algorithms can easily be extended to finer angular sampling (e.g. turns by 45° [17]). We also note that other approaches to turn detection (e.g., thresholding the magnitude of the data from the gyro [31]) have been proposed in the literature.

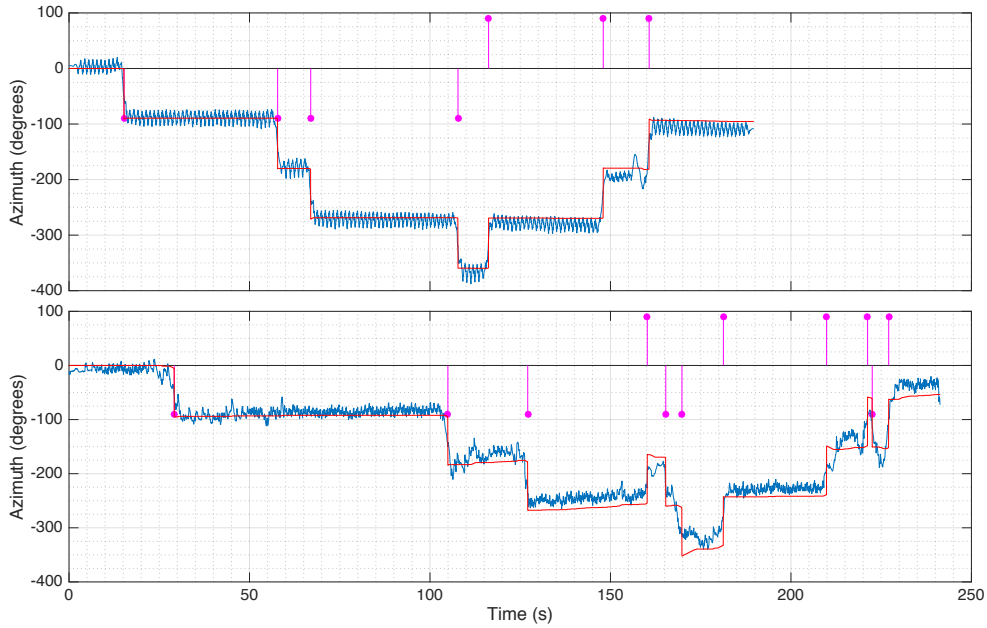


Fig. 13. The azimuth time series for two walkers (a sighted participant, top, and a blind participant using a cane, bottom) traversing identical paths. The pink stemmed circles represent turns by ±90° as detected by HMM-Turn 2. The red line represents the estimated azimuth $\hat{\alpha}_i$. Note that two sequences of 90° and −90° turns (overcounts) were incorrectly detected for the blind walker, at times of 170s and 242s. Some drift in the measured azimuth can be noted in both plots. For example, in the top plot (sighted walker), the drift is of approximately 8° per minute.

4.2.1.1 HMM-Turn 1. This algorithm uses a Hidden Markov Model (HMM) to represent a noisy sequence of azimuth observations. The states in the HMM represent the angular sector index k as defined above. For a given bias α_0, the Viterbi sequence of states can be easily computed from the time series of azimuth data with the constant value of α_0 removed from all measurements. Each state switch in the Viterbi sequence indicates a turn. The parameters of the HMM are:

- 4x4 transition probability matrix (A)
- Emission probability parameters (μ, σ)
- Prior state probabilities (π)

Note that the bias α_0 must be recomputed for each trial, because the reference frame used for the azimuth has arbitrary horizontal orientation (which may change each time the system is restarted). Given the HMM parameters, we run the Viterbi algorithm for multiple values of α_0 (between −50° and 50° with a stride of 5°), and keep the value of α_0 giving the highest likelihood of the associated Viterbi sequence.

The transition and prior state probabilities are learnt based on the measured relative frequencies of turns in the training data. The emission probability parameters are learnt by grid search, using the error metric defined later in Section 4.2.2 as criterion.

4.2.1.2 HMM-Turn 2. A drawback of HMM-Turn 1 is that it doesn't account for drift in the measured azimuth, which may occur due to time integration of any residual bias in the data from the gyro. Drift (which may be thought of as a slowly varying bias $alpha_0$) may accumulate during

a long path, potentially creating gross orientation errors. An example of slow drift is visible in Figure 13. Our HMM-Turn 2 algorithm explicitly tracks drift while also detecting turns. We give an overview of this algorithm in the following; for more details, the reader is referred to [17]. The idea is for the states in the HMM to represent the *difference* between the measured azimuth values at two consecutive time points, as due to either a variation bias or to a turn. More specifically, the state at any point represents one of three possible situations: (1) no change in heading; (2) a variation in bias; (3) a variation in heading due to a turn. Bias variations (state type 2) can take value equal to d or to $-d$, where d is an appropriate small constant learned from training data. An azimuth variation due to a turn (state type 3) is an angle of $k \cdot 90°$, where k takes value in $\{\pm 1, 2\}$. There are thus one state of type 1, two of type 2, and three of type 3 (totaling 6 possible states). An additional constraint is that if a state in the chain is of type 2 or 3, the next state in the chain must be of type 1. As a consequence, the 6x6 transition probability matrix for the Markov chain of states is forced to have only 6 non-null entries.

The HMM-Turn 2 algorithm computes a chain of states that best represents the observed data, in the following sense. Given a sequence of states $\{s_k\}$, we compute an estimated azimuth $\hat{\alpha}_i$ at the $i-th$ time sample as:

$$\hat{\alpha}_i = \alpha_0 + \sum\nolimits_{k \le i} f(s_k)$$

where $f(s_k) = 0$ for states of type 1, $f(s_k) = d$ or $-d$ for states of type 2, and $f(s_k) = 90°, -90°$, or $180°$ for states of type 3. The algorithm computes the sequence of states that minimizes the mean squared difference between the estimated azimuth sequence $\hat{\alpha}_i$ and the measured one. The algorithm's parameters are:

- 6 non-null entries of the transition probability matrix (A)
- Emission probability parameters (μ, σ)
- Prior state probabilities (π)
- Bias variation parameter (d)

As in the case of HMM-Turn 1, we estimate α_0 by post-facto grid search. Note that the bias variation parameter d, as well as the transition probabilities that lead to or from a state of type 2, are determined by grid search from the training data. (Unlike state of types 1 and 3, states of type 2 are not directly observable in the training data.)

An important observation concerning both algorithms is that computation of the optimal bias α_0 requires post-facto analysis of the whole sequence. If turn detection is to be performed on-line during a traversal, some other mechanism for the estimation of the bias α_0 would be needed. For example, one could estimate α_0 from the initial sequence of measurements, under the assumption that the user will walk straight (without turns) for at least a few seconds.

4.2.2 Error Metric. In order to assess the quality of our turn detection algorithms, we compute the text edit distance between the detected sequence of turns, and the ground-truth sequence. More precisely, we compute (using dynamic programming) the largest common subsequence of symbols, where each symbol represents a turn by a certain amount. Then, we count the rate of "undercounts" (turns that were not detected) and of "overcounts" (false positives), and divide them by the ground truth number of turns. The overall error (termed *TD-Error*) is equal to the sum of undercount and overcount rates. We believe that this error metric offers a more intuitive assessment of the turn detector behavior than the more complex "integral azimuth sequence" considered in our prior work [17].

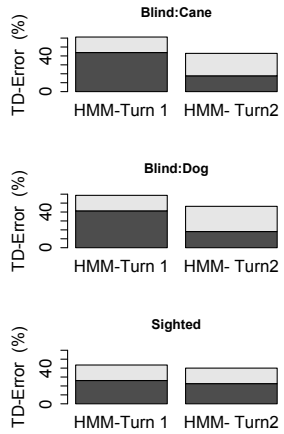


Fig. 14. Values of TD-Error averaged over participants for the different communities and different algorithms considered for the Stratified Leave-One-Out training modality. Gray bars indicate undercount rates, while black bars indicate overcount rates.

Note that our algorithms only considered turns by ±90° or 180°, while the paths also contained occasional 45° turns. (For example, in the sequence of paths traversed by participants P1-P8, there were 23 turns overall, three of which were 45° turns.) Hence, a 45° turn always produced an error event. More precisely, if an algorithm did not detect the 45° turn, an undercount event was recorded. If the algorithm detected the turn, assigning a 90° angle, both an undercount and an overcount event were recorded.

4.2.3 Turn Detection Results. The results of our turn detection algorithms are shown (in terms of undercount and overcount rates) in Figure 14. We performed a similar statistical analysis to the one presented for step counting algorithms (Section 4.1.3). The logarithm of TD-Error was used as dependent variable; the residuals of linear fit were confirmed Gaussian by the Shapiro-Wilk normality test.

4.2.3.1 Effect of Training. As in the case of the step counting algorithms, repeated measures ANOVA with blocking factors of *Participant*, *Algorithm*, and *Phone placement* found no significant difference in the mean value of TD-Error. We thus decided to concentrate on the *Stratified Leave-One-Out* modality in the following analysis.

4.2.3.2 Effect of Algorithm and Community. We tested the effect of *Algorithm* and *Community*, with blocking factors of *Participant* and *Phone placement*.

Community: The effect of Community did not quite reach the $\alpha = 0.05$ significance level (p=0.09). Multiple comparisons using Tukey's range test did show a significant difference in mean TD-Error between both *Blind:Cane* (52.0%) and *Sighted* (41.7%), as well as between *Blind:Dog* (52.5%) and *Sighted*.

Algorithm: It was found that *Algorithm* has a significant effect (p=1e-4) on the mean error. The average TD-Error of HMM-Turn 1 was 54.8%, compared to 42.6% for HMM-Turn 2.

No significant interaction between *Community* and *Algorithm* was found.

4.2.3.3 Effect of Phone Placement. As in the case of step counting, the location of the phones on our participants' garments did not show a significant effect on the average error.

4.2.3.4 Discussion. Both algorithms produced a relatively large error rate (42.6% for HMM-Turn 2, which was the better performing). As shown in Figure 14, HMM-Turn 1 gave a larger rate of overcounts than undercounts. This can be justified by the fact that HMM-Turn 1 is often affected by false positive pairs (e.g., an incorrectly detected turn by 90°, followed by a correction by −90°, resulting in a pair of overcount events). Two such false positive events are shown in Figure 13. HMM-Turn 2 produced a more balanced rate of undercount and overcounts (23% vs. 19%). In practice, this means that almost one out of four turns was missed by the algorithm, while approximately one out six turns detected by HMM-Turn 2 was a false positive.

The mean error recorded for sighted participants was 10% smaller than for blind participants; however, unlike in the case of step counting, this difference was not found to be statistically significant.

5 CONCLUSIONS

We have introduced a new data set with inertial sensor time series collected from ten blind walkers and five sighted walkers. Our participants walked through fairly long and complex routes; on their way, they sometimes had to open doors and avoid obstacles. Cumulatively, WeAllWalk contains inertial and other sensor data for a total of 7 miles traversed (approximately 20,000 annotated steps). The data has been subdivided into straight paths and turns, and carefully annotated, with special events (or features, such as bumping into an obstacle) individually identified and marked.

While we believe that this data can be useful to several researchers who are interested in personal mobility, we are also aware of some of its shortcomings. For example, although our participants were asked to walk naturally, they didn't have to find their way independently (as they were instructed when to turn). Participants may also have felt self-aware, as they were being followed and observed, and thus may not have been fully natural (for example, they may have put extra effort to avoid obstacles). All of our routes were indoors, and thus our data is not representative of outdoor ambulation. As one of our participants explained, some blind travelers pay attention to different things when walking indoors and outdoors. For example, when walking indoors, they may be careful to avoid obstacles such as a door left ajar; while walking outdoors, typical concerns include the condition of the pavement, and the possibility of a hole or a curb.

Based on the data in WeAllWalk, we have conducted two parallel analyses of algorithms for step counting and turn detection from inertial sensor data. Step counting is a widely used technique for approximate odometry; in combination with turn detection, it provides a simple localization strategy for walkers inside a building. A number of useful observations emerged from the comparative analysis of the algorithms' performance across the considered communities of walkers (blind walkers using a cane or a dog, and sighted walkers). This quantitative analysis may inform the design of localization and wayfinding systems that use inertial sensors, possibly integrated with other modalities (e.g., computer vision or iBeacons). More research is still needed to deal with complex situations, such as the automatic recognition of "unusual" situations (e.g., a walker negotiating an obstacle) that may result in inconsistent step counting or turn detection measurements. We believe that WeAllWalk, thanks to its fine-grain annotation and variety of walkers and of situations, could prove to be a precious resource even for this type of more complex analysis.

ACKNOWLEDGMENTS

Research supported by a Seed Grant from CITRIS, the Center for Information Technology Research in the Interest of Society. We would like to thank our participants for their help in building the WeAllWalk database.

REFERENCES

[1] [n. d.]. International Conference on Indoor Positioning and Indoor Navigation (IPIN). ipin-conference.org.

[2] [n. d.]. International Conference on Ubiquitous Positioning, Indoor Navigation and Location-Based Services (UPINLBS). upinlbs.sjtu.edu.cn.

[3] Dragan Ahmetovic, Cole Gleason, Chengxiong Ruan, Kris Kitani, Hironobu Takagi, and Chieko Asakawa. 2016. NavCog: a navigational cognitive assistant for the blind. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*. ACM, 90–99.

[4] Moustafa Alzantot and Moustafa Youssef. 2012. UPTIME: Ubiquitous pedestrian tracking using mobile phones. In *Wireless Communications and Networking Conference (WCNC), 2012 IEEE*. IEEE, 3204–3209.

[5] Michael Angermann, Patrick Robertson, Thomas Kemptner, and Mohammed Khider. 2010. A high precision reference data set for pedestrian navigation using foot-mounted inertial sensors. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*. IEEE, 1–6.

[6] Ilias Apostolopoulos, Daniel S Coming, and Eelke Folmer. 2015. Accuracy of pedometry on a head-mounted display. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 2153–2156.

[7] Stéphane Beauregard. 2007. Omnidirectional pedestrian navigation for first responders. In *Positioning, Navigation and Communication, 2007. WPNC'07. 4th Workshop on*. IEEE, 33–36.

[8] BB Blasch, SJ LaGrow, and WR De l'Aune. 1996. Three aspects of coverage provided by the long cane: Object, surface, and foot-placement preview. *Journal of Visual Impairment and Blindness* 90 (1996), 295–301.

[9] Agata Brajdic and Robert Harle. 2013. Walk detection and step counting on unconstrained smartphones. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. ACM, 225–234.

[10] Diansheng Chen, Wei Feng, Qiteng Zhao, Muhua Hu, and Tianmiao Wang. 2012. An infrastructure-free indoor navigation system for blind people. *Intelligent Robotics and Applications* (2012), 552–561.

[11] James Coughlan, Roberto Manduchi, and Huiying Shen. 2006. Cell phone-based wayfinding for the visually impaired. In *1st International Workshop on Mobile Vision*. 1–15.

[12] M Bernardine Dias, Aaron Steinfeld, and M Beatrice Dias. 2015. *Future directions in indoor navigation technology for blind travelers*. Boca Raton, FL: CRC Press.

[13] Marcus Edel and Enrico Köppe. 2015. An advanced method for pedestrian dead reckoning using BLSTM-RNNs. In *Indoor Positioning and Indoor Navigation (IPIN), 2015 International Conference on*. IEEE, 1–6.

[14] Navid Fallah, Ilias Apostolopoulos, Kostas Bekris, and Eelke Folmer. 2012. The user as a sensor: navigating users with visual impairments in indoor spaces using tactile landmarks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 425–432.

[15] Navid Fallah, Ilias Apostolopoulos, Kostas Bekris, and Eelke Folmer. 2013. Indoor human navigation systems: A survey. *Interacting with Computers* 25, 1 (2013), 21–33.

[16] German H Flores and Roberto Manduchi. 2016. WeAllWalk: An Annotated Data Set of Inertial Sensor Time Series from Blind Walkers. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*. ACM, 141–150.

[17] Germán H Flores, Roberto Manduchi, and Enrique D Zenteno. 2014. Ariadne's thread: Robust turn detection for path back-tracing using the iPhone. In *Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), 2014*. IEEE, 133–140.

[18] Randal C Foster, Lorraine M Lanningham-Foster, Chinmay Manohar, Shelly K McCrady, Lana J Nysse, Kenton R Kaufman, Denny J Padgett, and James A Levine. 2005. Precision and accuracy of an ankle-worn accelerometer-based pedometer in step counting and energy expenditure. *Preventive medicine* 41, 3 (2005), 778–783.

[19] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2001. *The elements of statistical learning*. Vol. 1. Springer series in statistics New York.

[20] Susan Vincent Graser, Robert P Pangrazi, and William J Vincent. 2007. Effects of placement, attachment, and weight classification on pedometer accuracy. *Journal of Physical Activity and Health* 4, 4 (2007), 359–369.

[21] Ann Hallemans, Els Ortibus, Francoise Meire, and Peter Aerts. 2010. Low vision affects dynamic stability of gait. *Gait & posture* 32, 4 (2010), 547–551.

[22] Michael Horvat, Christopher Ray, Vincent K Ramsey, Tanya Miszko, Roger Keeney, and Bruce B Blasch. 2003. Compensatory analysis and strategies for balance in individuals with visual impairments. *Journal of Visual Impairment and Blindness* 97, 11 (2003), 695–703.

[23] Rob Jackson. 2013. In what pocket do you keep your phone? http://phandroid.com/2013/03/25/in-what-pocket-do-you-keep-your-phone-poll/. (2013).

[24] Sampath Jayalath, Nimsiri Abhayasinghe, and Iain Murray. 2013. A gyroscope based accurate pedometer algorithm. In *International Conference on Indoor Positioning and Indoor Navigation*, Vol. 28. 31st.

[25] Amy A Kalia, Gordon E Legge, Rudrava Roy, and Advait Ogale. 2010. Assessment of indoor route-finding technology for people with visual impairment. *Journal of visual impairment & blindness* 104, 3 (2010), 135.

[26] Wonho Kang and Youngnam Han. 2015. SmartPDR: Smartphone-based pedestrian dead reckoning for indoor localization. *IEEE Sensors journal* 15, 5 (2015), 2906–2916.

[27] Jana Kosecka, Liang Zhou, Philip Barber, and Zoran Duric. 2003. Qualitative image based localization in indoors environments. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, Vol. 2. IEEE, II–II.

[28] Quentin Ladetto and Bertrand Merminod. 2002. In step with INS: Navigation for the blind, tracking emergency crews. *Gps World* 13, 10 (2002).

[29] Fan Li, Chunshui Zhao, Guanzhong Ding, Jian Gong, Chenxing Liu, and Feng Zhao. 2012. A reliable and accurate indoor localization method using phone inertial sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. ACM, 421–430.

[30] Andrea Mannini and Angelo Maria Sabatini. 2011. A hidden Markov model-based technique for gait segmentation using a foot-mounted gyroscope. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*. IEEE, 4369–4373.

[31] Juan Jose Marron, Miguel A Labrador, Adrian Menendez-Valle, Daniel Fernandez-Lanvin, and Martin Gonzalez-Rodriguez. 2016. Multi sensor system for pedestrian tracking and activity recognition in indoor environments. *International journal of ad hoc and ubiquitous computing* 23, 1-2 (2016), 3–23.

[32] Michael Marschollek, Mehmet Goevercin, Klaus-Hendrik Wolf, Bianying Song, Matthias Gietzelt, Reinhold Haux, and Elisabeth Steinhagen-Thiessen. 2008. A performance comparison of accelerometry-based step detection algorithms on a large, non-laboratory sample of healthy and mobility-impaired persons. In *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*. IEEE, 1319–1322.

[33] Kazuhiro Nakamura, Yoshiyuki Aono, and Yoshiaki Tadokoro. 1997. A walking navigation system for the blind. *Systems and computers in Japan* 28, 13 (1997), 36–45.

[34] Thanh Trung Ngo, Yasushi Makihara, Hajime Nagahara, Yasuhiro Mukaigawa, and Yasushi Yagi. 2014. The largest inertial sensor-based gait database and performance evaluation of gait-based personal authentication. *Pattern Recognition* 47, 1 (2014), 228–237.

[35] Bernd Pfrommer, Nitin Sanket, Kostas Daniilidis, and Jonas Cleveland. 2017. PennCOSYVIO: A Challenging Visual Inertial Odometry Benchmark. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 3847–3854.

[36] Bettina Pressl and Manfred Wieser. 2006. A computer-based navigation system tailored to the needs of blind people. *Computers Helping People with Special Needs* (2006), 1280–1286.

[37] Valérie Renaudin, Melania Susi, and Gérard Lachapelle. 2012. Step length estimation using handheld inertial sensors. *Sensors* 12, 7 (2012), 8507–8525.

[38] Timothy H Riehle, Shane M Anderson, Patrick A Lichter, William E Whalen, and Nicholas A Giudice. 2013. Indoor inertial waypoint navigation for the blind. In *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*. IEEE, 5187–5190.

[39] Angelo M Sabatini, Chiara Martelloni, Sergio Scapellato, and Filippo Cavallo. 2005. Assessment of walking features from foot inertial sensing. *IEEE Transactions on biomedical engineering* 52, 3 (2005), 486–494.

[40] Ronald W Schafer. 2011. What is a Savitzky-Golay filter?[lecture notes]. *IEEE Signal processing magazine* 28, 4 (2011), 111–117.

[41] Felix Scholkmann, Jens Boss, and Martin Wolf. 2012. An efficient algorithm for automatic peak detection in noisy periodic and quasi-periodic signals. *Algorithms* 5, 4 (2012), 588–603.

[42] Che-Chang Yang and Yeh-Liang Hsu. 2010. A review of accelerometry-based wearable motion detectors for physical activity monitoring. *Sensors* 10, 8 (2010), 7772–7788.

[43] Zheng Yang, Chenshu Wu, Zimu Zhou, Xinglin Zhang, Xu Wang, and Yunhao Liu. 2015. Mobility increases localizability: A survey on wireless indoor localization using inertial sensors. *ACM Computing Surveys (Csur)* 47, 3 (2015), 54.

[44] Mitsuru Yoneyama, Yosuke Kurihara, Kajiro Watanabe, and Hiroshi Mitoma. 2014. Accelerometry-Based Gait Analysis and Its Application to Parkinson's Disease Assessment?Part 1: Detection of Stride Event. *IEEE Transactions on neural systems and rehabilitation engineering* 22, 3 (2014), 613–622.

Received May 2017; revised August 2017; accepted November 2017