StopTracker: Real-time Monitoring of Visitor Stops and Preferences Using UWB Trajectory Streams

Fatima Hachem
fatme.hachem@unimi.it
University of Milan
Milan, Italy

Venkata Sudheer Siddabattula
srsudhir31@gmail.com
University of Milan
Milan, Italy

Davide Vecchia
davide.vecchia@unitn.it
University of Trento
Trento, Italy

Gian Pietro Picco
gianpietro.picco@unitn.it
University of Trento
Trento, Italy

Maria Luisa Damiani
maria.damiani@unimi.it
University of Milan
Milan, Italy

ABSTRACT

Real-time monitoring of visitor behavior in venues such as museums and shops offers valuable insights into visitor preferences, hotspots, and space occupancy. Visitors typically demonstrate interest, by stopping near points of interest (POIs), such as museum exhibits, for brief durations, often on the order of a few seconds, posing challenges for localization technology and stop detection. In this paper, we present *StopTracker*, a framework supporting the stop-move segmentation of indoor trajectory streams, relying on Ultra-wideband (UWB) localization for the accurate monitoring of visitor stops and preferences. StopTracker is constructed using $SPD_{streams}$, an advanced variant of the popular stop-detection technique *SPD*, and designed to handle *evolving stops*. Additionally, StopTracker support the dynamic ranking of POIs. By leveraging UWB technology, StopTracker aims to provide accurate and actionable insights into visitor behavior, enhancing both visitor experience and space management.

CCS CONCEPTS

• **Information systems → Spatial-temporal systems**; **Data streaming**.

KEYWORDS

Stop Detection, Ultra-wideband Localization, Trajectory Stream

ACM Reference Format:
Fatima Hachem, Venkata Sudheer Siddabattula, Davide Vecchia, Gian Pietro Picco, and Maria Luisa Damiani. 2024. StopTracker: Real-time Monitoring of Visitor Stops and Preferences Using UWB Trajectory Streams. In *The 32nd ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL '24), October 29-November 1, 2024, Atlanta, GA, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3678717.3691276

SIGSPATIAL '24, October 29-November 1, 2024, Atlanta, GA, USA

ACM ISBN 979-8-4007-1107-7/24/10
https://doi.org/10.1145/3678717.3691276

1 INTRODUCTION

The study of mobility behavior is a core topic in mobility data science [6], with applications ranging from urban planning and location-based recommendations to animal ecology. While outdoor behavior has been extensively studied using GNSS and check-in data, the analysis of indoor mobility data faces significant challenges due to the limited accuracy of commonly used positioning systems based on Bluetooth and Wi-Fi. In recent years, there has been growing interest in a new wave of positioning techniques that offer sub-meter accuracy. Among these, the Ultra-wideband (UWB) radio technology stands out due to its superior accuracy and potential applications in location tracking and proximity detection [11].

Mobility patterns and UWB. A widely observed mobility pattern is the *stop-move* pattern [7]: a *stop* (also known as a staying point or staying region) describes the permanence of an individual in a relatively small region, for a minimum amount of time; a *move* indicates a transition between stops. For example, stops may indicate places of interest, activities, events, which serve as natural proxies for individual behavior. Stop-move detection methods, initially designed for offline GNSS trajectories, typically partition trajectories into temporally disjoint segments alternating between stops and moves. These segments can be identified using various spatio-temporal criteria, such as speed, distance, point density, or statistical measures [4]. More recent techniques have been adapted for *symbolic trajectories* derived from mobile phone data [1]. Interestingly, recent work [3] provided quantitative evidence that state-of-the-art methods, such as Stay Point Detection (SPD) [5] and SeqScan [2], can be effectively applied to detect indoor stops in UWB trajectories, even when dealing with very brief stops lasting only a few seconds and occurring within a few decimeters of each other, as shown in Figure 1.

Stop-move over trajectory streams. The question we now focus on is whether these methods are suitable for the segmentation of trajectory streams in real-time applications, where a trajectory stream is an unbounded sequence of time-varying positions reporting the movement of an individual. Prompt detection of stops is useful in several circumstances, such as identifying overcrowded spaces where individuals tend to stop frequently or move slowly, detecting unsafe situations where stops may indicate suspicious behavior, reporting an emergency where stops may indicate a condition of danger. In all these cases, simply visualizing the trajectory stream, for example, through the use of time-varying heatmaps,

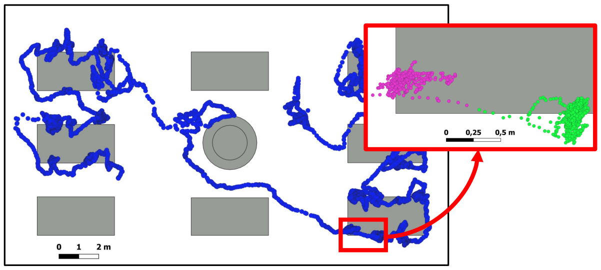


Figure 1: Example of a UWB trajectory in a museum. The area highlighted in red is zoomed in to show two closely spaced consecutive stops, detected using the SPD technique

is not sufficient to understand the meaning of the situation. In this paper, we leverage the availability of UWB movement data to present a case study regarding the analysis of visitors' movement in a museum, a challenging topic studied by social scientists since the mid-20th century. Visitors manifest their interest by stopping near installations for some time [8]. Timely identification of the visited POIs can provide museum curators, and potentially visitors as well, with real-time information on the attractiveness of the exhibition and space occupancy, which can be utilized in various ways. For instance, the presence of a crowd around an installation might suggest redirecting the flow of visitors.

Design guidelines and contributions. Detecting stop-move patterns in trajectory streams remains an underexplored area, with only a few exceptions [9]. In this work, we address the challenge of real-time applications, where stops evolve, and their status changes over time. We propose a design principle for stop-move segmentation in trajectory streams: *the spatio-temporal conditions defining a stop should be continuously monitored, the detection of a stop should be notified as soon as these conditions are satisfied, and the stop should be continuously updated until it ends.* Accordingly, we define a stop as follows: a stop is *started* when the minimal spatio-temporal conditions for a segment to be a stop, are met; it is *active* when the existing stop is expanded; and finally is *closed* when the stop is ended. In this work, we begin to analyze the suitability of state-of-the-art methods for detecting stops in trajectory streams. We focus in particular on the popular *SPD* method [5], due to its simplicity and efficiency. We demonstrate that *SPD* does not satisfy the aforementioned design principle. Motivated by this observation, we introduce a variant of the algorithm, $SPD_{streams}$, and present an application, *StopTracker*, for the continuous monitoring of visitor stops and preferences using UWB trajectories streams. The rest of the paper is organized as follows: we present $SPD_{streams}$ in Section 2 and StopTracker in Section 3. Section 4 outlines the demo showing StopTracker at work in the museum. Open directions will be briefly discussed in the conclusive section.

2 REVISING THE SPD METHOD

The *SPD* algorithm processes the timestamped positions in the trajectory sequentially. A trajectory segment is identified as a stop when the spatial distance between the segment starting point and ending point exceeds a threshold δ, and the segment time duration

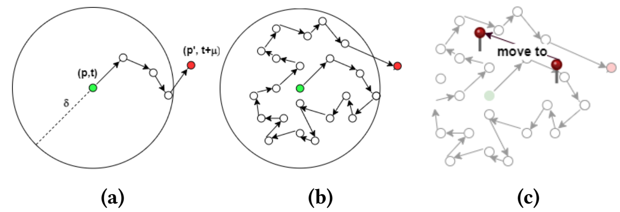


Figure 2: (a) *SPD*: a stop is recognized when the distance from the beginning of the segment (green dot) exceeds δ (red dot); (b) therefore a long permanence in the region may cause a long delay; (c) $SPD_{streams}$: the stop of minimum duration is identified and then expanded, while its centroid may shift

is greater than ρ. More specifically, the segment starting at (p, t) and ending at $(p', t + \mu)$ is a stop if p' is the first point in the sequence, such that the distance $dist(p, p') > \delta$ and $\mu \geq \rho$ (Figure 2 (a)). As a result, a stop can only be detected once it has already concluded. Therefore, if an individual remains in a region for an extended period (see Figure 2 (b)), the stop is recognized with a significant delay. Such delays are impractical when real-time or immediate reporting of stops is required. To address this issue, we propose $SPD_{streams}$.

$SPD_{streams}$ implements the aforementioned design principle. The algorithm continuously monitors the temporal condition, while ensuring that the distance between a potential starting point p and the current point p_c remains within a threshold δ. Once the temporal distance exceeds ρ, a stop is identified, with p as the starting point, covering a time span from p to p_c. The centroid is then calculated as the average of the points between p and p_c. The stop may be further expanded: for each subsequent point, if its distance to p remains within δ, the stop incorporates the new point, and both the centroid and duration are updated accordingly (see Figure 2 (c)). Whenever the distance exceeds δ, the stop is closed, and the new point becomes the potential starting point of a future stop. Conversely, if a stop cannot be generated because the distance exceeds δ but the temporal distance is still less than ρ, the initial point is updated, and the process continues with subsequent points. The detailed algorithm is reported in Algorithm 1 .

3 THE STOPTRACKER SYSTEM

$SPD_{streams}$ is the core component of the StopTracker application. *StopTracker* inspects the input trajectory streams to dynamically provide visual information on the evolving stops, highlighting as well the POIs preferred by visitors on the basis of historical data. In the following, we describe two primary functionalities of the system: POI detection and dynamic ranking. Subsequently, we outline the architecture of the prototypical system.

POI detection. StopTracker determines the POIs in proximity to which visitors stop, and dynamically compute the duration of those stops. As a spatio-temporal point is read, the system determines whether a stop of minimum duration is created. If a stop is created, the system attempts to match the stop centroid to a POI. This matching process identifies the closest POI within a distance threshold R from the stop centroid. If no POI exists within this threshold, the matching result is *null*, indicating that the visitor stopped at a location not associated with a POI. It is assumed that the association of a stop with a POI (or null) remains constant over time. However, an existing stop may expand if additional spatio-temporal points are added to it. Therefore, the duration of the stop must be updated from its initial creation time.

Dynamic ranking. Visitor preferences are dynamically determined based on two measures: (i) the *total duration* of stops (both closed and open) at each POI from all visitors up to time t, and (ii) the *number of distinct visitors* stopping at each POI up to time t. Using these measures, we introduce a method for assigning a time-varying score in the range [0,1] to POIs, depending on the desired granularity. For the sake of flexibility, we assume that POIs can be aggregated in *groups*, based for example on proximity or other properties. Consider a set of POI groups $G_1, \ldots, G_n$. Let $X_j(t)$ the total duration of the stops at the POIs of the group G_j at time t; similarly, $Y_j(t)$ the total number of visitors stopping at the POIs in G_j. And let $X_{min}(t), X_{max}(t), Y_{min}(t), Y_{max}(t)$ the minimum/maximum duration and number of visitors among the n groups at time t. Each group G_j is assigned a score $S_j(t)$ obtained from the linear combination of the min-max normalized $X_j(t)$ and $Y_j(t)$ variables.

$$S_j(t) = \alpha \frac{X_j(t) - X_{min}(t)}{X_{max}(t) - X_{min}(t)} + (1-\alpha) \frac{Y_j(t) - Y_{min}(t)}{Y_{max}(t) - Y_{min}(t)}$$

with $\alpha \in [0, 1]$, the trade-off parameter. A value of $\alpha = 1$ indicates that the score is computed solely based on the total duration, while with $\alpha = 0$, the computation is based on the number of visitors. The resulting scores are then visually presented by highlighting their ranking.

Proof-of-concept. In this demo, we utilize data streams generated from existing UWB trajectories. The *StopTracker* architecture in Figure 3 (top), consists of three main components: a Trajectory Stream Generator, a Processing Server, and a Dashboard. The *Trajectory Stream Generator* emulates the collection of movement data for multiple individuals in real time. It is implemented to read pre-recorded UWB positions from an input file, where each row corresponds to a specific point for an individual, including its location and timestamp. These points are transmitted to the *Processing Server* sequentially, accounting for the time intervals between consecutive points to precisely replicate the real-time sequence of events. The Processing Server processes the spatio-temporal points of each individual on a separate thread, which includes a Stop Manager as described in Figure 3 (bottom). The *Stop Manager* detects stops using $SPD_{streams}$, monitors the status of stops for each received point, and communicates this information to the *Dashboard* for visualization. When a stop starts, the Stop Manager also checks for the nearest POI and updates the details of active stops to maintain accurate POI rankings. These updates are sent to the Dashboard as well. The Dashboard provides functionalities for configuring the system and visualizing the individual positions, stops and their evolution over space and time. It also displays dynamic updates on rankings.

4 OUTLINE OF THE DEMO

The demo shows StopTracker at work in a major science museum in Italy (MUSE, Trento). The target area contains six tables hosting 36 exhibits that vary in size and content, including text panels, interactive elements, and screens. Additionally, the space contains three benches. The trajectory stream is generated from the real trajectories collected from a sample of 10 visitors, upon their prior consent. Figure 4 displays the area and shows one of the visitors stopping at an exhibit. These trajectories have been acquired through the TALLA system [10], via time-difference-of-arrival (TDoA) localization, using an UWB infrastructure consisting of 10 anchors deployed on the ceiling of a $25 \times 15m^2$ area. Each visitor wears a necklace with a UWB tag on the chest, a natural option for a real use. The position of the tag is the one actually recorded by TALLA. Positions are sampled at a frequency of 12Hz. The visits in the area last about 10 minutes. The stop detected in these trajectories were validated

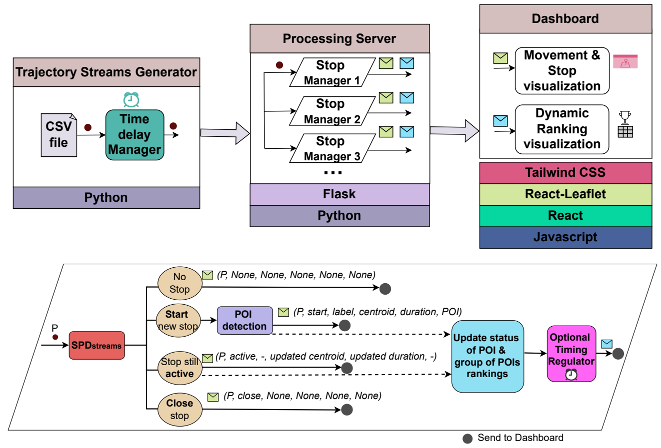
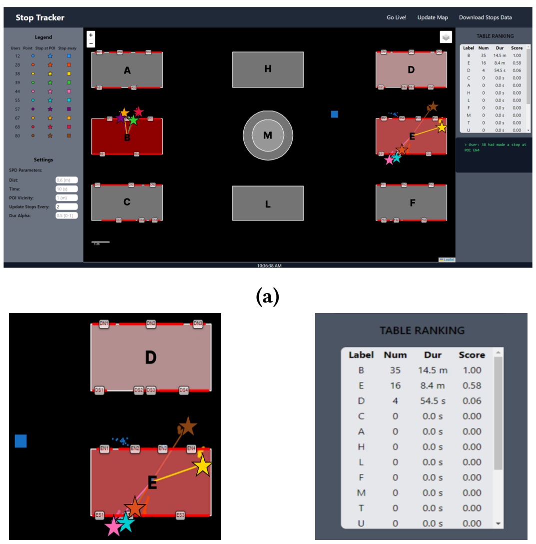


Figure 3: (Top) Overview of the StopTracker architecture, including the Trajectory Stream Generator, the Processing Server and the Dashboard, and the underlying software layers; (bottom) focus on the Stop Manager structure

Algorithm 1: $SPD_{streams}$ for one point stream

```
Input: points stream, δ and ρ
current_stop ← None, i ← 1
while IS_STREAMING do
    status ← None
    j ← i + 1
    while IS_STREAMING do
        if distance(p_i, p_j) < δ then
            if current_stop == None then
                if ΔTime(p_i, p_j) > ρ then
                    status ← start
                    current_stop ← create_new_stop
            else
                status ← active
                expand_current_stop
            j ← j + 1
        else
            if current_stop == None then
                i ← reassess_initial_point
                j ← j + 1
            else
                status ← close
                close_current_stop
                i ← j
                break
        send_notification_to_Dashboard
```

Figure 4: A visitor stopping at an exhibit in the target area of the case study, where the pinpoint indicates the stop location. The area contains six long tables hosting exhibits



Label	Num	Dur	Score
B	35	14.5 m	1.00
E	16	8.4 m	0.58
D	4	54.5 s	0.06
C	0	0.0 s	0.00
A	0	0.0 s	0.00
H	0	0.0 s	0.00
L	0	0.0 s	0.00
F	0	0.0 s	0.00
M	0	0.0 s	0.00
T	0	0.0 s	0.00
U	0	0.0 s	0.00

(a)

(b) **(c)**

Figure 5: (a) Overview of the Dashboard. The central part is the dynamic map of the target area. (b) Zoom on current stops: five visitors are stopping at table E. (c) Zoom on ranking: the dynamic scores of tables. Table B is the top ranked

against ground truth data derived from video recordings of the visitors' experience, showing high spatial and temporal accuracy.

4.1 Interacting with StopTracker

Figure 5 (a) illustrates the Dashboard and the information displayed during the processing of the trajectory stream. The central part of the Dashboard is the *map*: it shows the layout of the target area in the museum, including the tables [A-F], the POIs as labelled linear elements, and the benches (H, M, L). This map dynamically displays various pieces of information, organized in layers, including: the partial traces (most recent positions) of each visitor, the open stops (i.e., started, active), the total number of visitors and the duration of stops at every POI, and the visitor preferences aggregated at the level of table. Additionally, through the interface, users can configure the system parameters (*settings*), which include the $SPD_{streams}$ parameters, the threshold distance R, and the α parameter. Figure 5 (b) highlights the different types of active stops in the map. Stops are represented by their centroids and comprise those associated with a POI (a star symbol); and the null stops (a square symbol). The map shows that 4 users are stopping at some POI hosted by table B, 5 users at table E, and 1 user stopping far apart from POIs. The map also shows the last positions of visitors (dots). The localization error is in on the order of a few decimeters. Figure 5 (c) details the scores computed for the groups of POIs located on each table. Tables are then ranked based on this score and displayed with different colors to highlight their relative position in the ranking. For example, table E appears to be more attractive than table D at the time the map is displayed. In the demo setup, the map is continuously updated.

5 CONCLUSION

This demo focuses on real-time stop detection in high-accuracy trajectory streams. We highlight the opportunities offered by UWB-based localization in decision support systems, an aspect that has been largely overlooked by the scientific community despite the growing popularity of commercial solutions. Furthermore, we emphasize the importance of cross-disciplinary research, particularly at the intersection of IoT and mobility data science. This is critical for developing scalable architectures tailored to crowded indoor environments and enabling advanced visitor analytics.

ACKNOWLEDGMENTS

This work was supported in part by project SERICS (PE00000014) under the NRRP MUR program funded by the EU - NGEU. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the Italian MUR. Neither the European Union nor the Italian MUR can be held responsible for them. This work is also partially supported by the PNRR ICSC National Research Centre for High Performance Computing, Big Data and Quantum Computing (CN00000013), under the NRRP MUR program funded by the NextGenerationEU.

REFERENCES

[1] M.L. Damiani, F. Hachem, C. Quadri, M. Rossini, et al. 2020. On Location Relevance and Diversity in Human Mobility Data. *ACM Trans. Spatial Algorithms Syst.* 7, 2, Article 7 (2020), 38 pages.

[2] M. L. Damiani, F. Hachem, H. Issa, N. Ranc, et al. 2018. Cluster-based Trajectory Segmentation with Local Noise. *Data Mining and Knowledge Discovery* 32 (2018), 1017–1055.

[3] F. Hachem, D. Vecchia, M.L. Damiani, and G.P. Picco. 2022. Fine-grained Stop-Move Detection in UWB-based Trajectories. In *Proc. of IEEE International Conference on Pervasive Computing and Communications (PerCom).*

[4] S. Hwang, C. Vandemark, N. Dhatt, S.V. Yalla, et al. 2018. Segmenting human trajectory data by movement states while addressing signal loss and signal noise. *International Journal of Geographical Information Science* 32 (2018).

[5] Q. Li, Y. Zheng, X. Xie, Y. Chen, et al. 2008. Mining user similarity based on location history. In *Proc. ACM SISGPATIAL.*

[6] M. Mokbel, M. Sakr, L. Xiong, A. Züfle, et al. 2024. Mobility Data Science: Perspectives and Challenges. *ACM Transactions on Spatial Algorithms and Systems* 10, 2, Article 10 (2024), 35 pages.

[7] C. Parent, S. Spaccapietra, C. Renso, G. Andrienko, et al. 2013. Semantic trajectories modeling and analysis. *ACM Computing Surveys* 45, 4 (2013), 32.

[8] B. Serrell. 1998. *Paying Attention: Visitors and Museum Exhibitions.* American Alliance Of Museums.

[9] R. Uddin, M.T. Mahin, P. Rajan, Chinya R., et al. 2023. Dwell Regions: Generalized Stay Regions for Streaming and Archival Trajectory Data. *ACM Transactions on Spatial Algorithms and Systems* 9, 2, Article 9 (2023), 35 pages.

[10] D. Vecchia, P. Corbalán, T. Istomin, and G. P. Picco. 2019. TALLA: Large-scale TDoA Localization with Ultra-wideband Radios. In *Proc. of International Conference on Indoor Positioning and Indoor Navigation (IPIN).*

[11] F. Zafari, A. Gkelias, and K. K. Leung. 2019. A Survey of Indoor Localization Systems and Technologies. *IEEE Communications Surveys & Tutorials* 21, 3 (2019), 2568–2599.