Supplementary Information for
Pedestrian mobility citizen science complements expert mapping for enhancing inclusive neighborhood placemaking

Ferran Larroya[1,2,*], Roger Paez[3], Manuela Valtchanova[3], and Josep Perelló[1,2,**]

[1]OpenSystems Research Group, Departament de Física de la Matèria Condensada, Universitat de Barcelona, Martí i Franquès, 1, 08028, Barcelona, Catalonia, Spain
[2]Universitat de Barcelona Institute of Complex Systems, Martí i Franquès, 1, 08028, Barcelona, Catalonia, Spain
[3]ELISAVA Barcelona School of Design and Engineering, Universitat de Vic- Universitat Central de Catalunya, La Rambla, 30-32, 08002 Barcelona, Catalonia, Spain
[*]ferran.larroya@ub.edu
[**]josep.perello@ub.edu

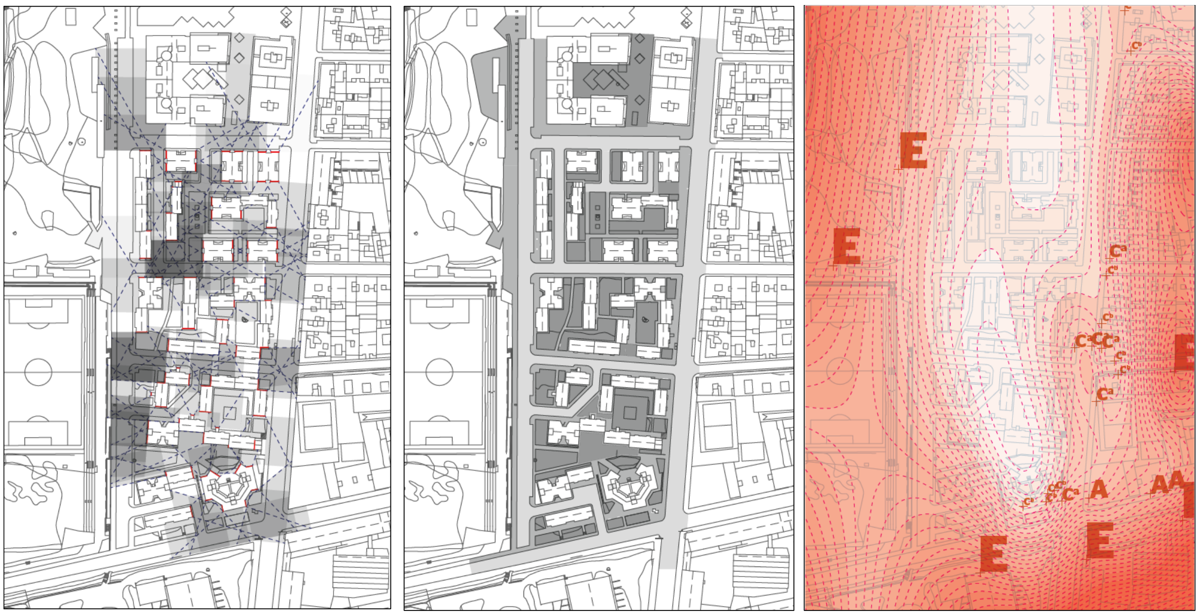
Supplementary Notes

List of Supplementary Figures

List of Supplementary Tables

1 Opportunity Units

Figure S1 shows the mappings developed from a visual, spatial and socio-cultural approach that allowed the expert architects to obtain the eight opportunity units (OU) [1, 2]. Figure 2 in the main text shows the result of the overlay of the three high activation potential maps, resulting in the eight OU. Details of the procedure are described in the main paper.



(a) (b) (c)

Figure S1: Mapping procedure to determine the activation opportunity units. (a) Visual dimension: Result of the interaction between urban objects in the neighborhood and the degree of visual vertical space generated by the blind facades of the buildings. The darker regions (where several visibility areas converge) are considered regions of potential interest. (b) Spatial dimension: There are four levels of activation depending on the degree of pedestrian accessibility of the public space: from zero activation (private space, in white) to high activation (intermediate spaces like squares and green flowerbeds, in 70% black). (c) Social dimension: Topographic map with the attractors of socio-cultural activity in the neighborhood. The socio-cultural agents are considered attractors, and are assigned a certain height based on the degree of affluence of people they attract. There are three main groups: Education and cultural locations (high level of affluence), local agents (medium level of affluence) and local stores (high or low affluence).

1.1 Mean stop duration across expert-defined opportunity units

To complement Table 2 in the main text, which reports the number of stops, total stopping time and descriptive statistics for each opportunity unit (OU), Figure S2 presents the mean stop duration ($\langle\tau\rangle$) with its standard error of the mean (SE) for each OU. The results reveal substantial heterogeneity in mean stop duration across OUs. Units P01, P03 and P04 do not contain action stops, and therefore show null values. Among the remaining units, C03 exhibits the highest mean stop duration; however, this value corresponds to a single stop and should therefore be interpreted cautiously. C01 and C04 display comparatively long average stopping times combined with larger variability, whereas C02 and P02 present shorter average durations.

These differences in mean stop duration do not necessarily mirror the number of stops per OU, reinforcing the need to consider both frequency and duration when characterizing the distribution of action stops across opportunity units.

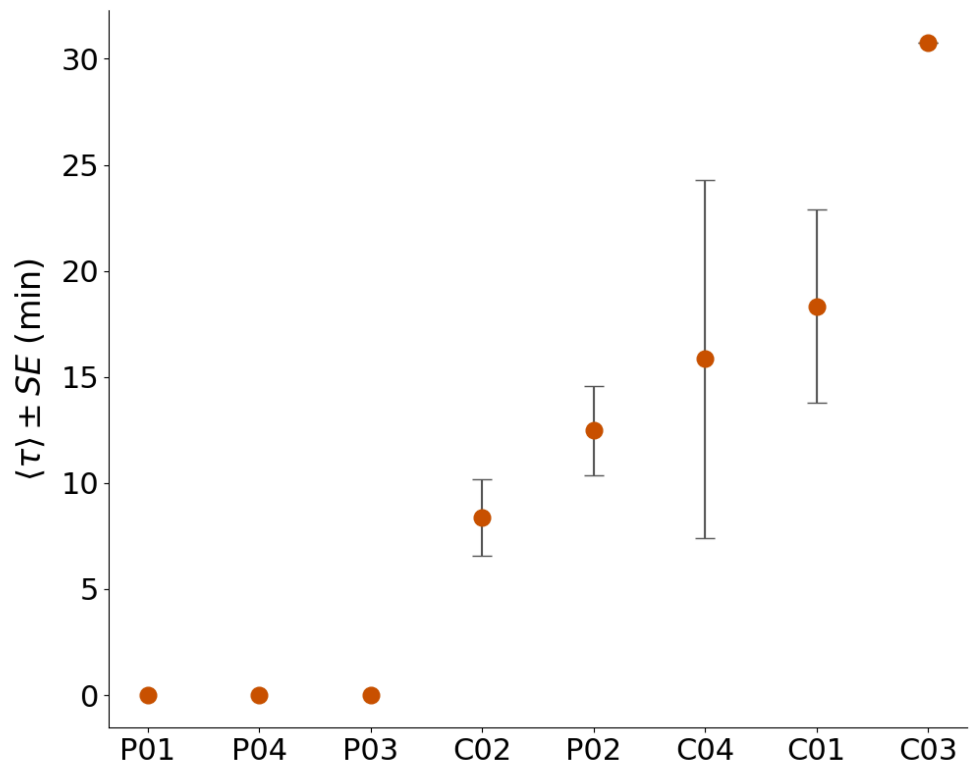
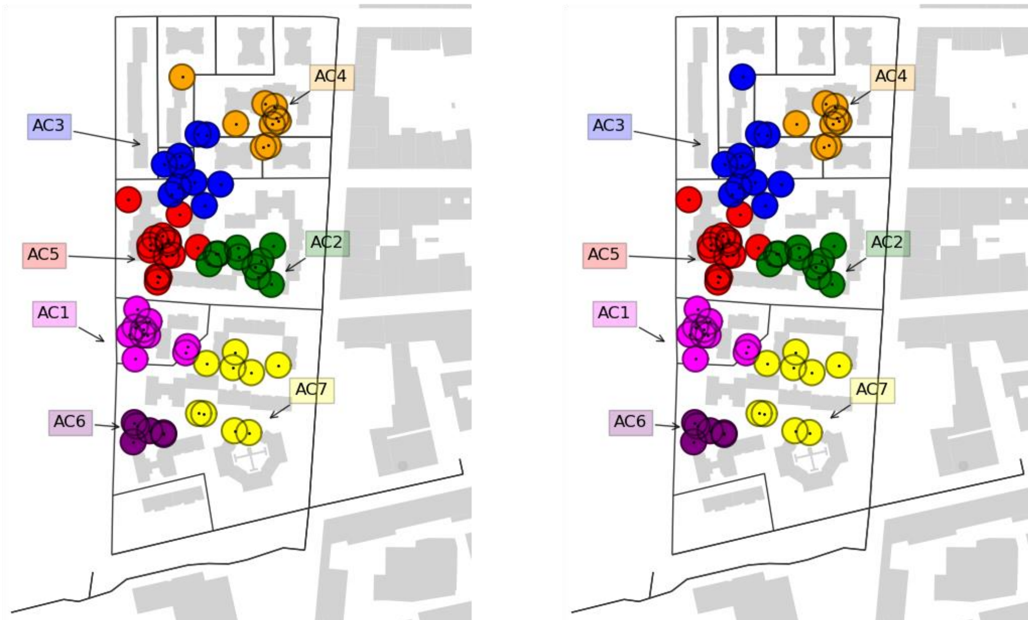


Figure S2: Mean stop duration across expert-defined opportunity units (OUs). Average stop duration ($\langle\tau\rangle$) with standard error of the mean (SE) for action stops occurring within each OU. Units without detected action stops show null values. This figure complements Table 2 in the main manuscript by providing a visual comparison of stop duration across opportunity units.

2 K-means clustering

2.1 Unweighted K-means

Using the classic K-means algorithm [3, 4] (unweighted), we obtain essentially the same clusters as using the weighted version, in which each stop is weighted with its duration. Figure S3 compares the cluster representation in both cases. We observe that there is only one stop (the one located at the top of the map) that has changed from the AC3 activation cluster (in the weighted version, Figure S3a) to AC4 (in the unweighted version, Figure S3b). Table S1 displays the statistics of the stops of each activation cluster in the unweighted version. The statistics are essentially the same as in the weighted K-means (see Table 3 in the main paper), and there are only minor differences in the AC3 and AC4 clusters which exchange one stop.



(a) (b)

Figure S3: Action stops within the 7 K-means clusters. The 7 activation clusters applying the K-means algorithm. (a) Weighted version, using the time duration of each stop as the weights. (b) Unweighted version. For visualization purposes, all stops are represented with bubbles of the same size, with each cluster of a different color.

Cluster	Stops (%)	Total T_{stop} (s)	$\langle T_{stop}\rangle$ (s)	$\sigma_{T_{stop}}$ (s)	T^{min}_{stop} (s)	T^{max}_{stop} (s)
AC1 (magenta)	10 (15%)	$8,083$ (15%)	808 ± 202	638	100	$1,955$
AC2 (green)	10 (15%)	$10,532$ (20%)	$1,053 \pm 276$	872	200	$3,130$
AC3 (blue)	11 (15%)	$12,305$ (22%)	$1,118 \pm 252$	834	96	$2,677$
AC4 (orange)	8 (13%)	$4,480$ (9%)	560 ± 117	332	100	963
AC5 (red)	14 (20%)	$7,761$ (15%)	547 ± 150	563	87	$2,345$
AC6 (purple)	6 (9%)	$4,349$ (8%)	725 ± 162	397	262	$1,243$
AC7 (yellow)	9 (13%)	$5,650$ (11%)	628 ± 262	785	39	$2,669$
All stops	68 (100%)	$53,060$ (100%)	780 ± 83	680	39	$3,130$

Table S1: Statistics of the 7 K-means unweighted activation clusters. Number of stops, total duration of stops, average stop duration, standard deviation and the shortest and the longest stop for each one of the 7 K-means activation clusters (unweighted case).

2.2 Optimal K number of clusters: Elbow, Silhouette and GAP statistic methods

The first metric we compute to study quality of the resulting clusters and therefore the number of optimal clusters is the within-cluster sum of squares (WCSS, Eq. (4) from the main paper), known as Elbow method [5]. Figure S4 shows the evolution of WCSS as a function of K. Although the elbow point is not neatly observed, the shape of the curve suggests that a value of K between 5 and 7 could be a reasonable optimal value.

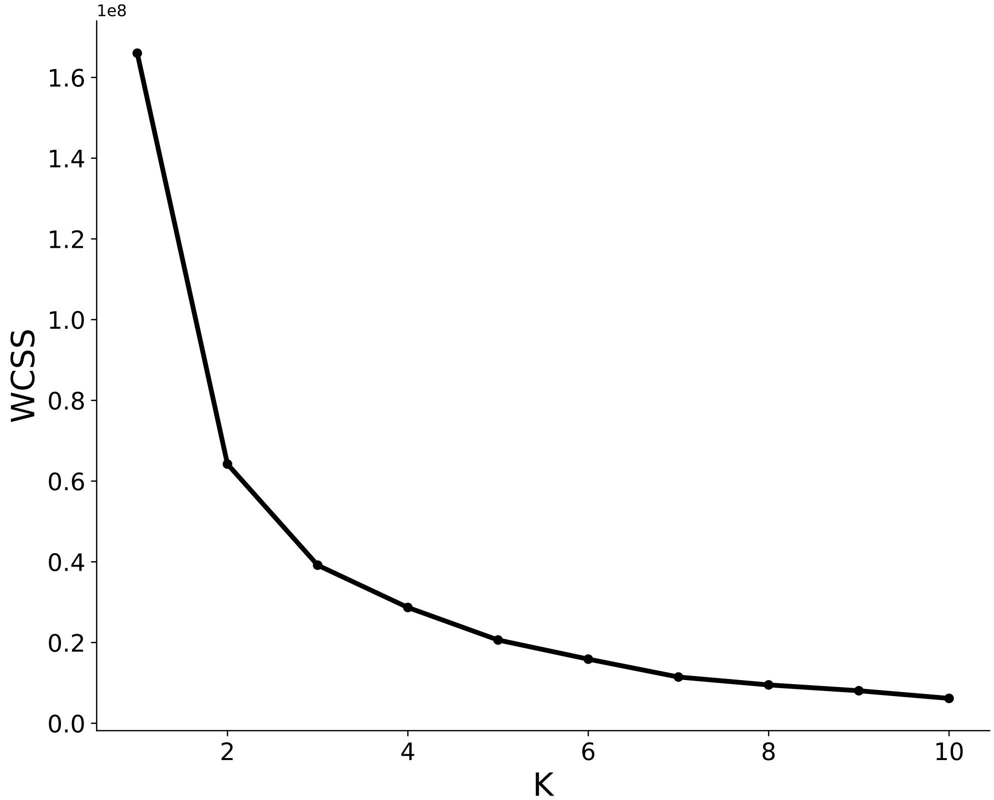


Figure S4: Within-cluster sum of squares as a function of the number of clusters. The shape of the within-cluster sum of squares (WCSS) curve shows that a value of K between 5 and 7 could be a reasonable choice for the number of optimal clusters.

We also study the Silhouette coefficients [6] (s_i), which measure the separation between the resulting clusters, i.e., how close each point in a cluster is to the points in the neighboring clusters. Figure S5 shows that $K = 7$ is the number of clusters that maximize the averaged Silhouette coefficients ($\langle s_i \rangle$), and therefore is the optimal K.

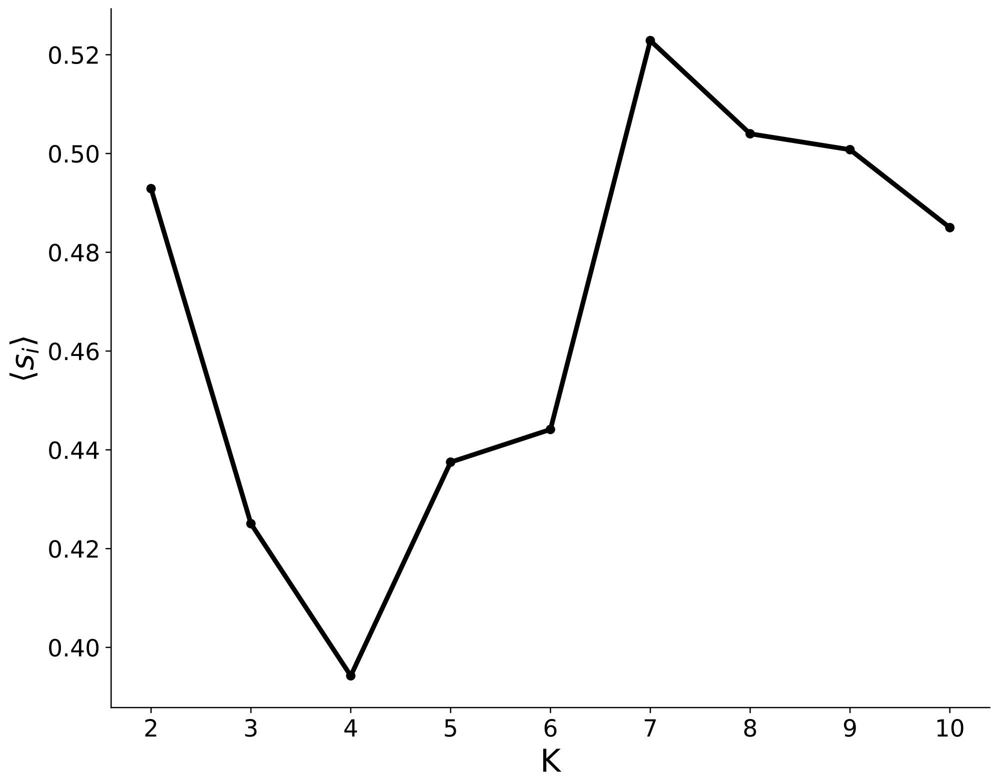
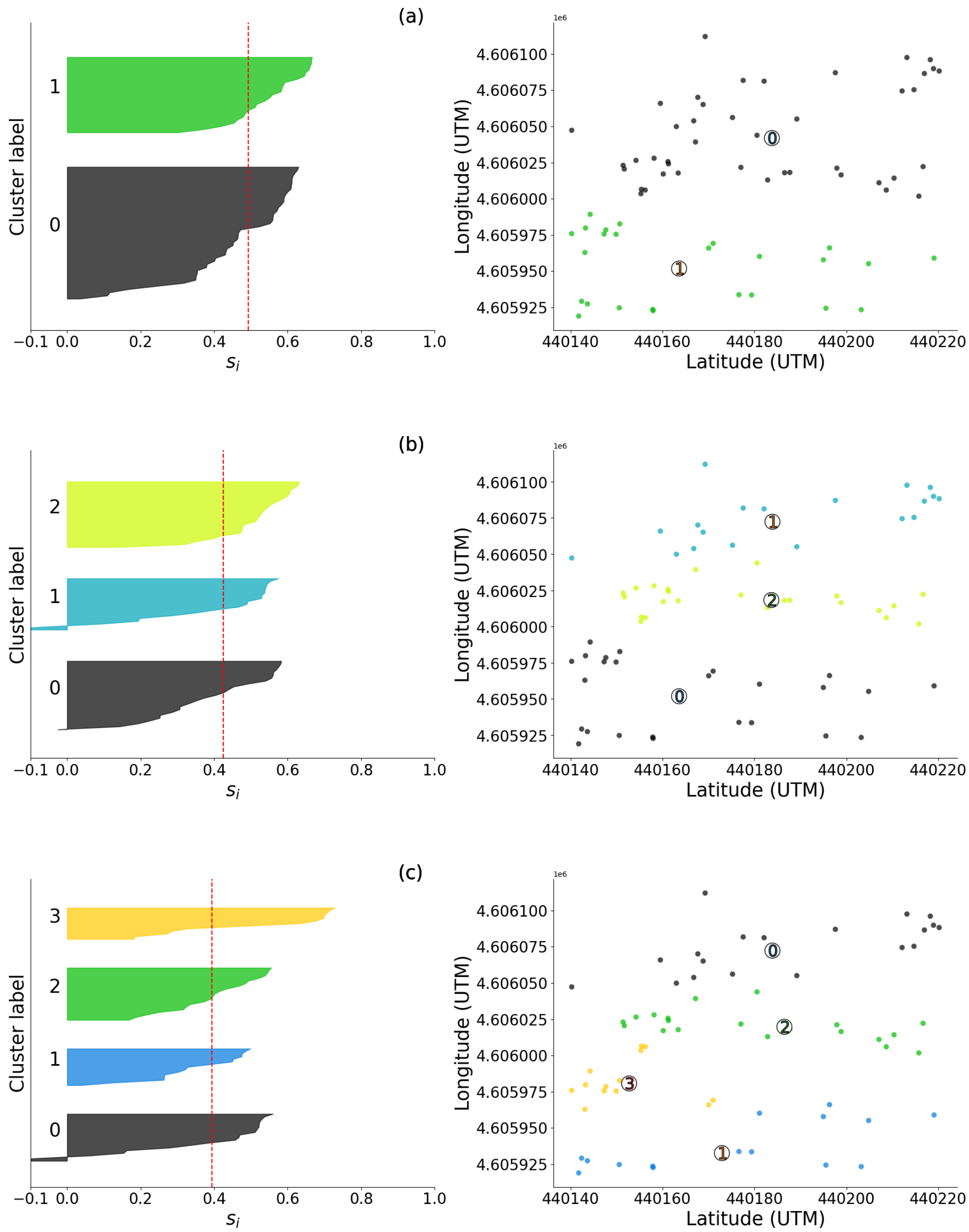
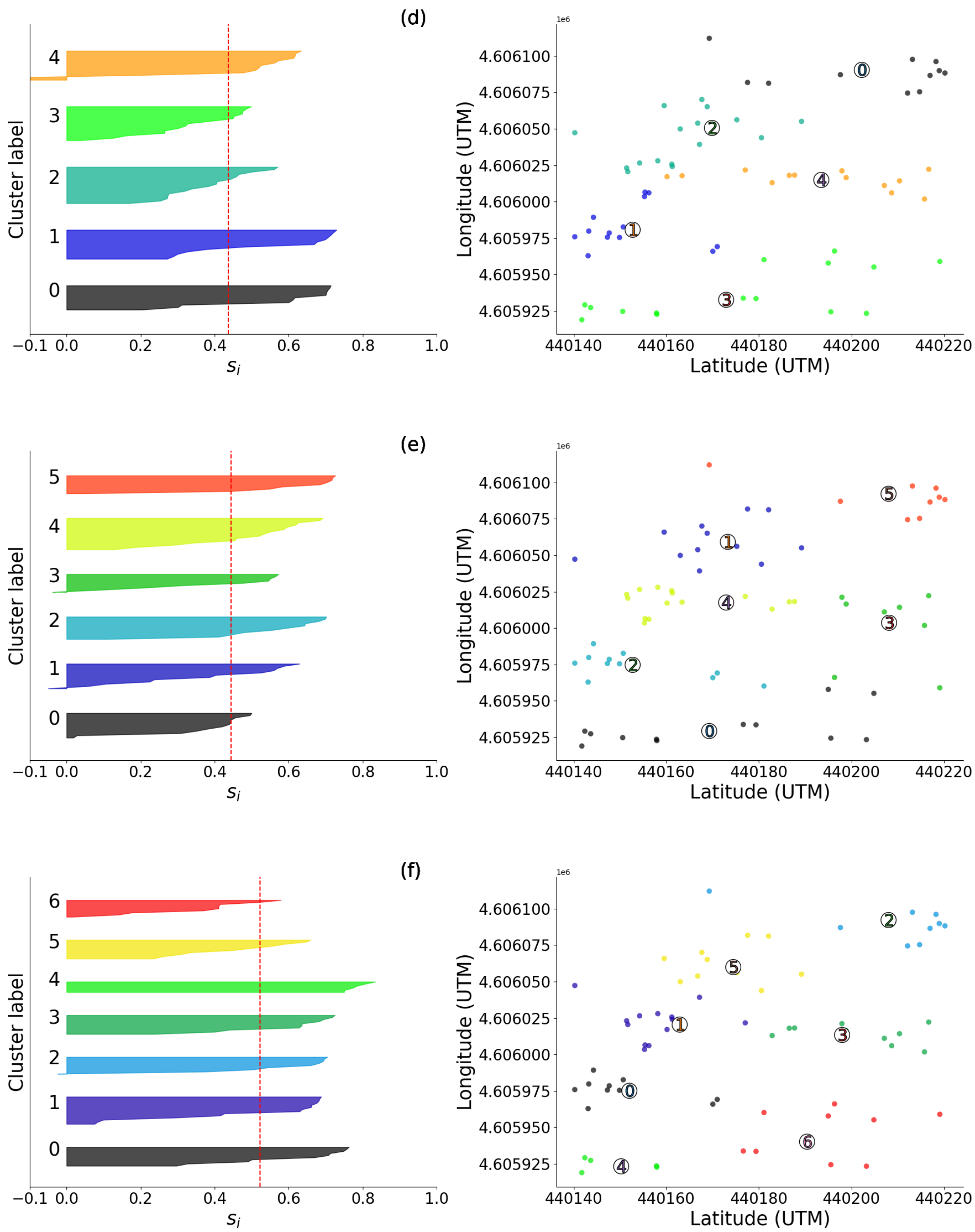


Figure S5: Averaged Silhouette value as a function of the number of clusters. The Silhouette coefficients $\langle s_i \rangle$ represent the separation between clusters. The optimal number of clusters K is chosen as the one that maximizes the averaged Silhouette coefficients over all data points ($K = 7$ in this case).

Figure S6 shows 9 panels (from $K = 2$ to $K = 10$) with the individual Silhouette coefficients and the averaged Silhouette scored over all data points, together with a cluster representation of the data points. We confirm that $K = 7$ is the optimal choice for the number of clusters, since the Silhouette coefficients are quite uniform and close to the average value, with practically no negative values.


(a)
Cluster label
1
0
−0.1 0.0 0.2 0.4 0.6 0.8 1.0
s_i
1e6
4.606100
4.606075
4.606050
4.606025
4.606000
4.605975
4.605950
4.605925
Longitude (UTM)
440140 440160 440180 440200 440220
Latitude (UTM)
0
1
(b)
Cluster label
2
1
0
−0.1 0.0 0.2 0.4 0.6 0.8 1.0
s_i
1e6
4.606100
4.606075
4.606050
4.606025
4.606000
4.605975
4.605950
4.605925
Longitude (UTM)
440140 440160 440180 440200 440220
Latitude (UTM)
1
2
0
(c)
Cluster label
3
2
1
0
−0.1 0.0 0.2 0.4 0.6 0.8 1.0
s_i
1e6
4.606100
4.606075
4.606050
4.606025
4.606000
4.605975
4.605950
4.605925
Longitude (UTM)
440140 440160 440180 440200 440220
Latitude (UTM)
0
2
3
1


(d)
(e)
(f)
Cluster label
s_i
−0.1 0.0 0.2 0.4 0.6 0.8 1.0
Longitude (UTM)
Latitude (UTM)
1e6
4.606100
4.606075
4.606050
4.606025
4.606000
4.605975
4.605950
4.605925
440140 440160 440180 440200 440220

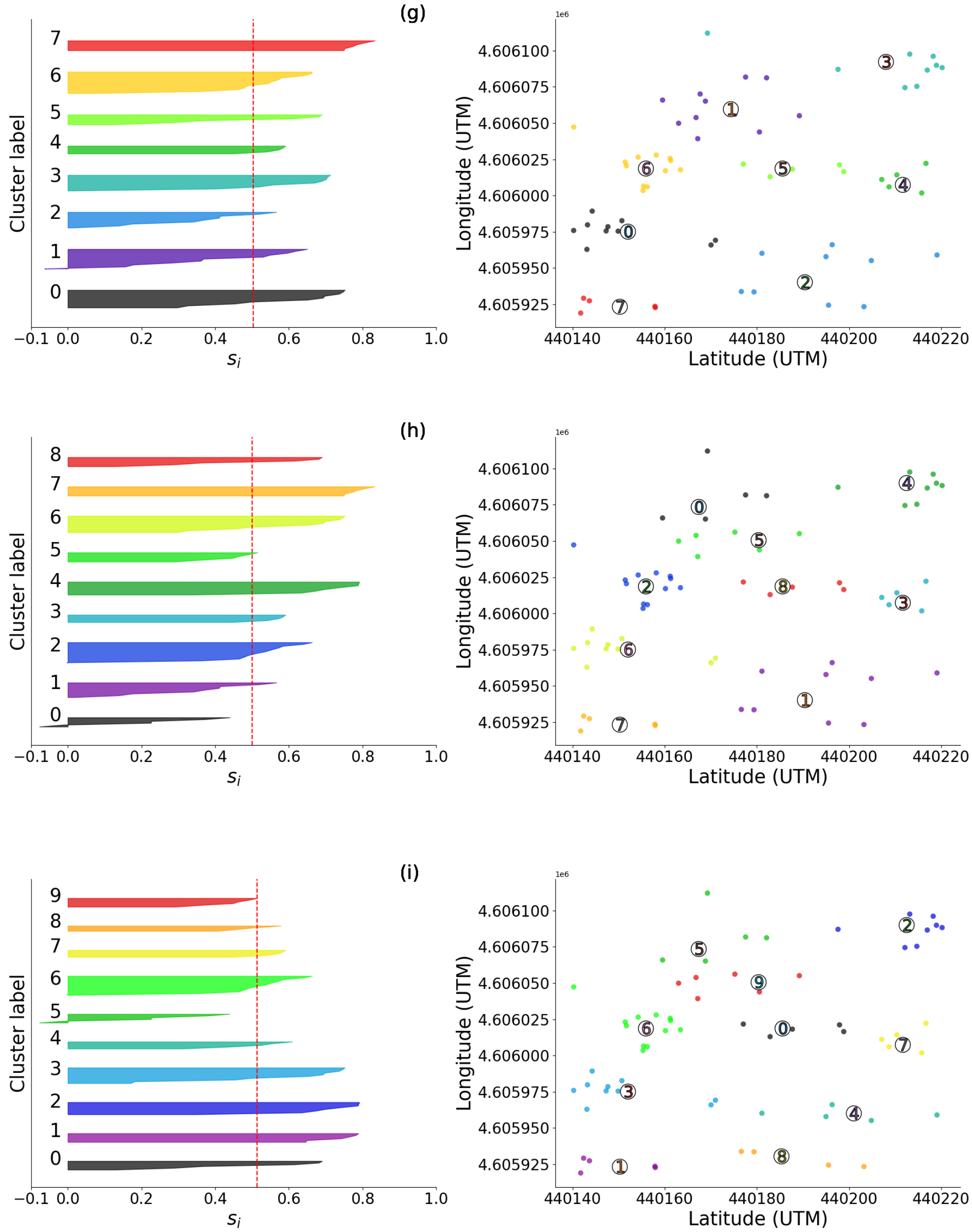


Figure S6: Individual Silhouette coefficients of each cluster (left panels) and cluster representation (right panels). (a) $K = 2$. (b) $K = 3$. (c) $K = 4$. (d) $K = 5$. (e) $K = 6$. (f) $K = 7$. (g) $K = 8$. (h) $K = 9$. (i) $K = 10$. The cluster representation is done using the UTM projection of the latitude and the longitude. The red vertical dotted line is the averaged Silhouette score over all data points ($\langle s_i \rangle$). We check that $K = 7$ is the optimal choice not only because the averaged Silhouette score $\langle s_i \rangle$ is maximized but also because most of the coefficients are above (or near) the average value, with practically no negative values.

Finally, using the GAP statistic method [7] we also study how the clustering structure is far from a random uniform distribution of data points, comparing the within-cluster sum of squares (WCSS). The optimal number of clusters is the smallest K such the GAP(K) is maximized, within one standard deviation of the GAP($K+1$). Figure S7 shows the evolution of the GAP statistic with K. Again, we find $K=7$ to be the number of optimal clusters.

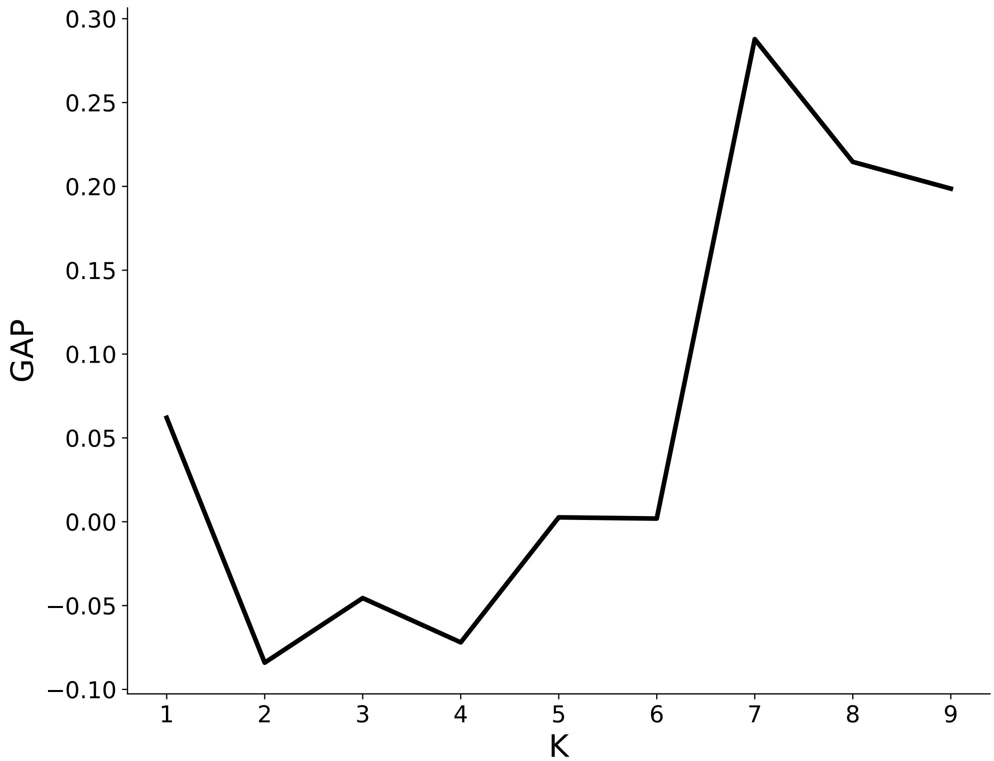


Figure S7: GAP statistic as a function of the number of clusters. The GAP value measures how the clustering structure is far away from the random uniform distribution of data points. The optimal number of clusters K is then chosen to be smallest one that maximizes the GAP(K), i.e., is within one standard deviation of the GAP($K+1$). In our case, the optimal number of clusters is $K=7$.

2.3 DBSCAN clustering

We also explored other algorithms to classify stops in high-density regions. In particular, we applied a well-known density-based algorithm called DBSCAN [8]. The main difference is that DBSCAN automatically determines clusters based on the density of data points, whereas K-means is a centroid-based clustering (least squares optimization). DBSCAN receives two parameters, min_{points} and ϵ, and the number of clusters does not need to be specified before applying the algorithm. It works as follows:

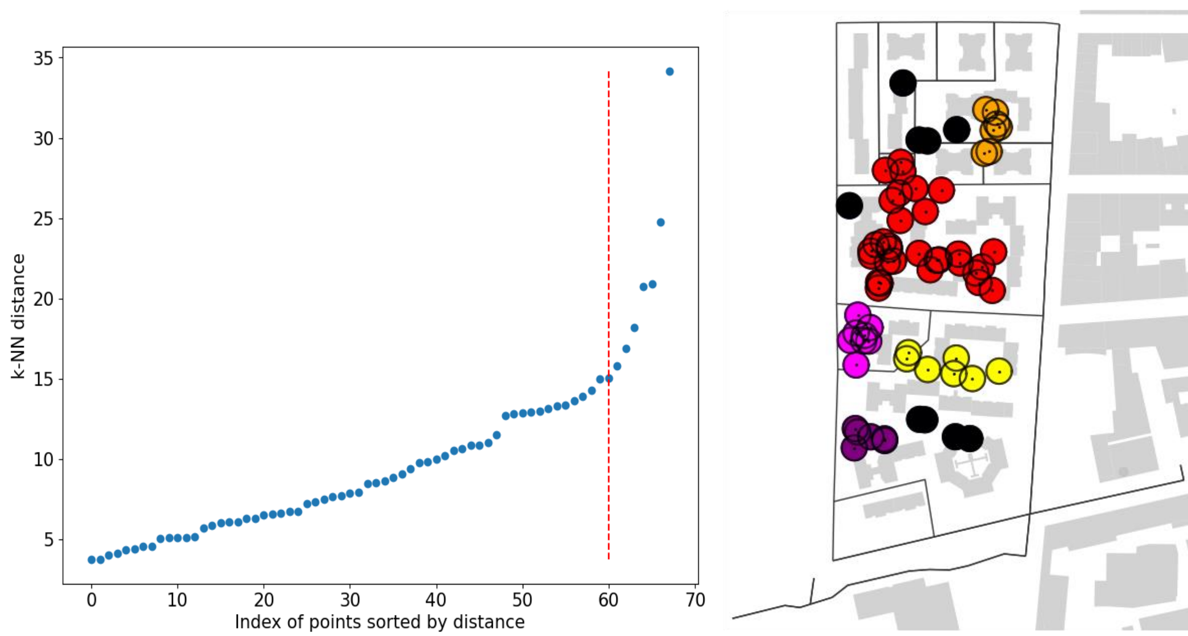
1. Computes the distances between each pair of points
2. For each point (P) it computes how many neighbors it has, using the parameter ϵ as the maximum distance to consider two points as neighbors. Therefore, considering another point, Q, if $dist(P,Q) < \epsilon$, then they are neighbors.
3. The algorithm receives another parameter called min_{points}, which is the minimum number of points, neighbors of P, to consider P as a core element (high-density sample).
4. The clusters are formed iterating over all points, and considering the neighboring points that are core elements.

5. If another point, say M, is not a core point but is a neighbor of P and can be connected through a chain of core points (for example, going from P to M trough Q), then M is also considered as part of the cluster (border).

6. The rest of the points, which are neither boundaries of the cluster nor core points, are classified noise.

Although it is not necessary to predefine the number of clusters, the algorithm is very sensitive to the choice of both parameters, especially the threshold ϵ. Similar to the Elbow method, to determine the optimal value of ϵ, it is useful to use a k-distance plot. The idea is to calculate the average distance of each point from its k-nearest neighbors. Next, the k-distances are plotted in ascending order. The aim is to determine the "knee", which is the threshold where a sharp change occurs along the k-distance curve and corresponds to the optimal value of ϵ. For the parameter min_{points}, it is very often to use 4 with 2-D data-sets ($min_{points} = 2D$ for a data-set with D dimensions).

Figure S8a shows the k-distance plot with a vertical line corresponding to the index value where a knee is located. The k-distance corresponding to that value is the optimal ϵ, which is $\epsilon = 15.07m$. The knee is obtained with a method from a library called kneebow (https://github.com/georg-un/kneebow), which uses a Rotor to simply rotate the data, so that the curve looks down and then take the minimum value (elbow). Running the DBSCAN algorithm with $\epsilon = 15.07$ and $min_{data} = 4$, 5 clusters are found with 9 points (stops) classified as noise. Figure S8b shows the map representation of the 5 clusters (black points correspond to noise).



(a) (b)

Figure S8: DBSCAN density-based algorithm to classify the pausing events. (a) k-distance plot to determine the optimal value the parameter ϵ to run the DBSCAN algorithm. The vertical red dotted line corresponds to the value where the knee of the curve is found. (b) The map representation of the 5 clusters found automatically by the algorithm, with 9 points labelled as noise (in black).

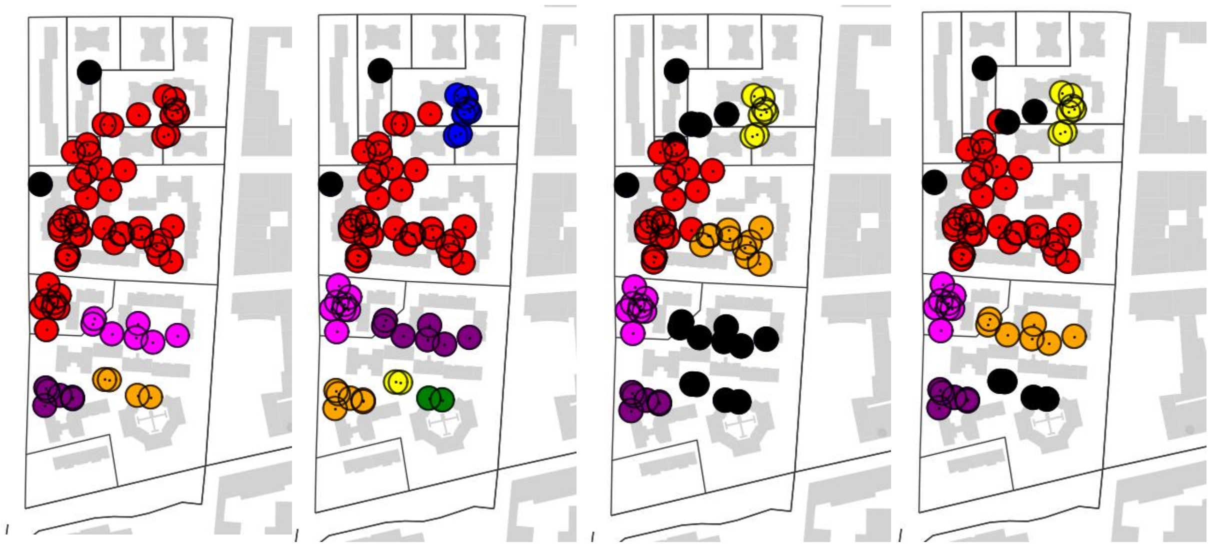
In Table S2 we show the results (number of clusters and noisy records found by the DBSCAN algorithm) for different combinations of the parameters ϵ and $\min_{data}$. As can be expected, as ϵ increases (larger distances for the threshold of neighbors), the number of clusters is reduced, so the algorithm may be capturing more than one high-density region in a single cluster. On contrary, for smaller values of ϵ, the condition to be neighbors is very strict, so there are many clusters and many noisy stops. For a fixed ϵ, changing $\min_{points}$ means changing the minimum number of neighboring points above which we consider a stop as a high density point (core element). Therefore, increasing $\min_{point}$ leads to fewer clusters and more noisy points.

ϵ \ min_{points}	2	3	4	5	6
10	15 clusters 12 noisy points	9 clusters 24 noisy points	5 clusters 37 noisy points	4 clusters 43 noisy points	2 clusters 53 noisy points
12	11 clusters 8 noisy points	7 clusters 16 noisy points	6 clusters 21 noisy points	6 clusters 24 noisy points	5 clusters 33 noisy points
15	8 clusters 3 noisy points	5 clusters 9 noisy points	5 clusters 9 noisy points	4 clusters 16 noisy points	5 clusters 17 noisy points
18	7 clusters 2 noisy points	5 clusters 6 noisy points	5 clusters 8 noisy points	5 clusters 10 noisy points	4 clusters 15 noisy points
20	4 clusters 2 noisy points	4 clusters 2 noisy points	3 clusters 6 noisy points	4 clusters 8 noisy points	3 clusters 13 noisy points
25	2 clusters 1 noisy point	2 clusters 1 noisy point	2 clusters 1 noisy point	2 clusters 2 noisy points	2 clusters 4 noisy points

Table S2: Number of clusters and noisy points for different model parameters. Number of clusters and noisy points obtained for different combinations of DBSCAN model parameters, ϵ and min_{points}. Smaller values of ϵ than 15 (the optimal found), lead to a huge amount of noisy punts. For greater values, there are few noisy records but the number of clusters is also small (larger distances for the threshold of neighbors imply larger groups). As can be expected, increasing min_{points} leads to few number of clusters.

Figure S9 shows four clustering representations for different combinations of ϵ and min_{points}. We observe that DBSCAN can not separate the big red cluster as the K-means algorithm does. Both Figure S9a and Figure S9b have only 2 noisy records, but in the first case the algorithm is not able to distinguish different high-density regions (big red cluster) and in the other case DBSCAN is overestimating some high-density region (green and yellow clusters). Figure S9c is an example of a very noisy result with a high min_{points} and Figure S9d is similar to Figure S8 using the optimal value of ϵ.

We believe that K-means is a better option to capture and distinguish the high-density regions of actions on this high-resolution scale, mainly because DBSCAN it is not able to classify in different clusters the large number of stops in the central part of the neighborhood.



(a) (b) (c) (d)

Figure S9: DBSCAN clustering results for different model parameters. (a) For $\epsilon = 20$ and $min_{points} = 3$, the algorithm found 4 clusters and 2 noisy points. (b) For $\epsilon = 18$ and $min_{points} = 2$, the algorithm found 7 clusters and 2 noisy points. (c) For $\epsilon = 15$ and $min_{points} = 6$, the algorithm found 5 clusters and 17 noisy points. (d) For $\epsilon = 18$ and $min_{points} = 4$, the algorithm found 5 clusters and 8 noisy points.

2.4 Cluster stability and robustness analysis

We have assessed the robustness and stability of the clustering results [9, 10], employing tests to evaluate whether the detected spatial clustering configuration remains consistent under variations in data sampling and positional uncertainty.

Sub-sampling stability. We performed repeated sub-sampling of 80% of the action stops without replacement. For each iteration, the weighted K-means clustering (with fixed K) was recomputed and compared to the reference clustering obtained using the full dataset. This procedure was repeated 200 times. Similarity between cluster partitions was quantified using the Adjusted Rand Index (ARI, Eq. (10) from the main paper) [11], which measures the degree of pairwise agreement between two clustering partitions. In addition to ARI, we quantified centroid stability by computing the displacement between corresponding centroids in the reference and perturbed clusterings.

Figure S10a shows the distribution of ARI values under sub-sampling, while Figure S10b reports the corresponding average centroid displacements. For $K = 7$, the mean ARI is 0.865 (standard deviation $SD = 0.074$), with more than 95% of the iterations yielding an ARI greater than or equal to 0.73. The mean centroid displacement is $5.13\ m$ ($SD = 2.4\ m$), with 95% of the displacements below $8\ m$. These results indicate a high degree of structural stability under data reduction. Notably, centroid shifts remain small relative to the spatial extent of the neighborhood, reinforcing the robustness of the detected clusters.

Gaussian positional perturbation. To evaluate robustness against GPS measurement uncertainty, we added isotropic Gaussian noise to the projected spatial coordinates

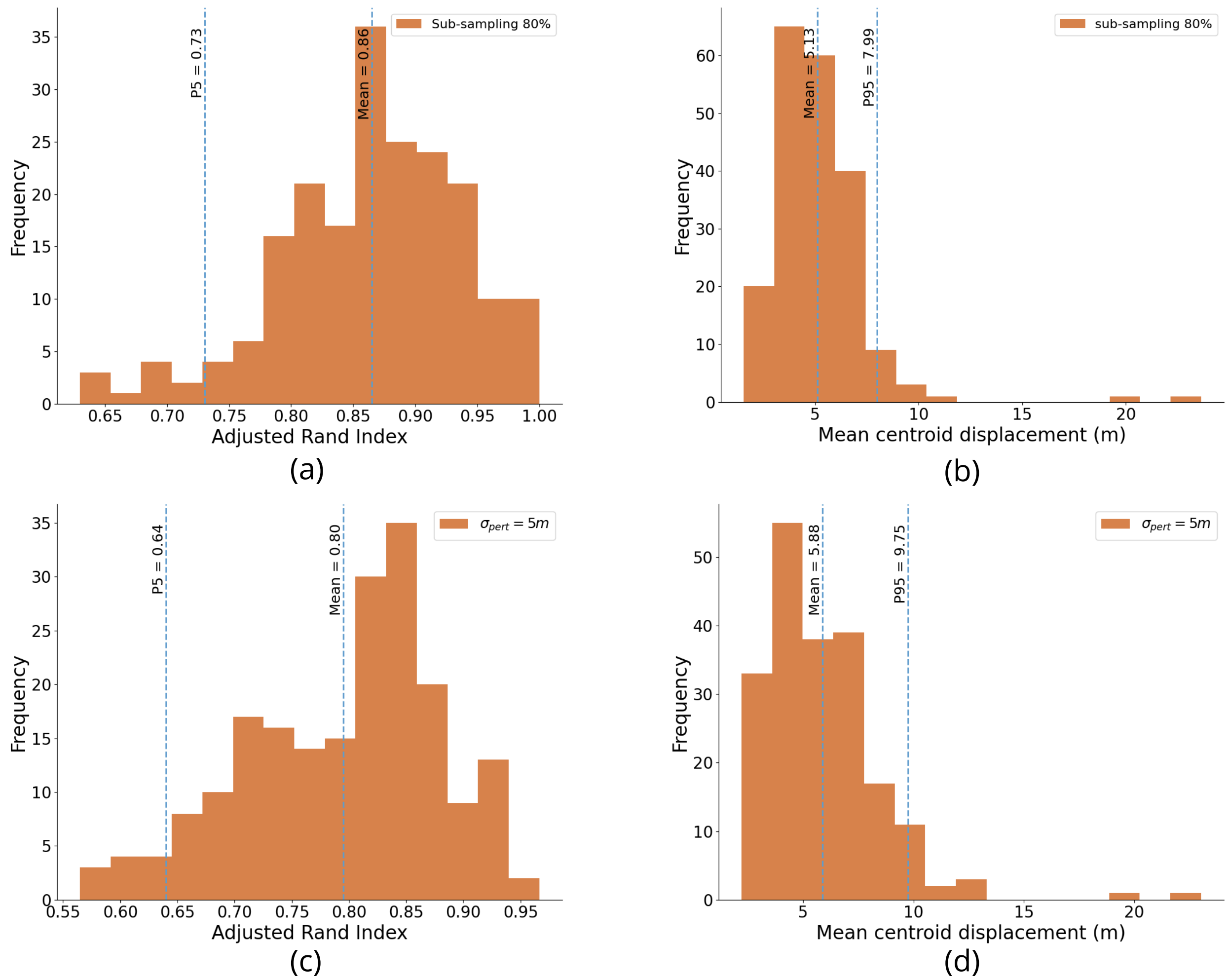


Figure S10: Cluster stability under sub-sampling and positional perturbation. (a) Distribution of the Adjusted Rand Index (ARI) obtained from 200 iterations of 80% sub-sampling without replacement. (b) Distribution of the corresponding mean centroid displacement (in meters) under sub-sampling. (c) Distribution of ARI values obtained from 200 iterations of Gaussian positional perturbation ($\sigma = 5$ m). (d) Distribution of the corresponding mean centroid displacement under Gaussian perturbation. Vertical dashed lines indicate the mean value and selected percentiles for each distribution. These results quantify the structural and geometric stability of the clustering solution.

$$x' = x + \epsilon_x, \quad y' = y + \epsilon_y, \tag{1}$$

where $\epsilon_x, \epsilon_y \sim \mathcal{N}(0, \sigma^2)$ and $\sigma = 5$ m, corresponding to typical horizontal positional uncertainties (on the order of a few meters) reported for consumer-grade GPS devices in urban environments [12]. For each perturbed dataset, weighted K-means clustering was recomputed and compared to the reference solution using the ARI and centroid displacement metrics. This procedure was repeated 200 times.

Figures S10c–d show the distributions of ARI and average centroid displacement under positional perturbation. For $K = 7$, the mean ARI remains high (0.795, $SD = 0.084$), with more than 95% of the iterations yielding an ARI greater than or equal to 0.64. The mean centroid displacement is 5.88 m ($SD = 2.6$ m), with 95% of the displacements below 10 m. These results indicate that the spatial partition remains stable under realistic levels of positional measurement noise.

Co-assignment matrix stability. We also computed the co-assignment probability matrix across iterations [13], which measures the proportion of iterations in which both stops were assigned to the same cluster.

For $K = 7$, the mean co-assignment probability for pairs belonging to the same reference cluster is 0.90 under sub-sampling and 0.91 under Gaussian perturbation. In contrast, the probability for pairs belonging to different clusters remains below 0.02 in both cases. The strong separation between within-cluster and between-cluster co-assignment probabilities indicates a highly stable and well-separated partition.

Comparison of alternative values of K. To complement the Elbow, Silhouette and GAP statistic selection of the optimal number of clusters, we visually compared the spatial partitions obtained for $K = 6$, $K = 7$ and $K = 8$ (Figure S11). The three solutions share a very similar spatial structure, indicating that the clustering pattern is robust to small variations in K.

For $K = 6$ (Figure S11a), two adjacent activation clusters (AC5 in red and AC3 in blue) are merged into a single larger region, while the remaining clusters remain essentially unchanged relative to the $K = 7$ partition (Figure S11b). This aggregation reduces spatial resolution by combining areas that appear as distinct high-density zones when $K = 7$ is used. In contrast, $K = 8$ (Figure S11c) introduces a small subdivision at the interface between AC5 (red) and AC2 (green), creating a minor central cluster. This additional split does not reveal a clearly differentiated spatial pattern and mainly increases fragmentation. Therefore, solutions for $K = 6$ and $K = 8$ retain the same fundamental spatial structure, differing only by a mild aggregation or subdivision of adjacent areas. Within this stable framework, $K = 7$ provides a balanced level of spatial resolution, avoiding both unnecessary merging and minor over-segmentation.

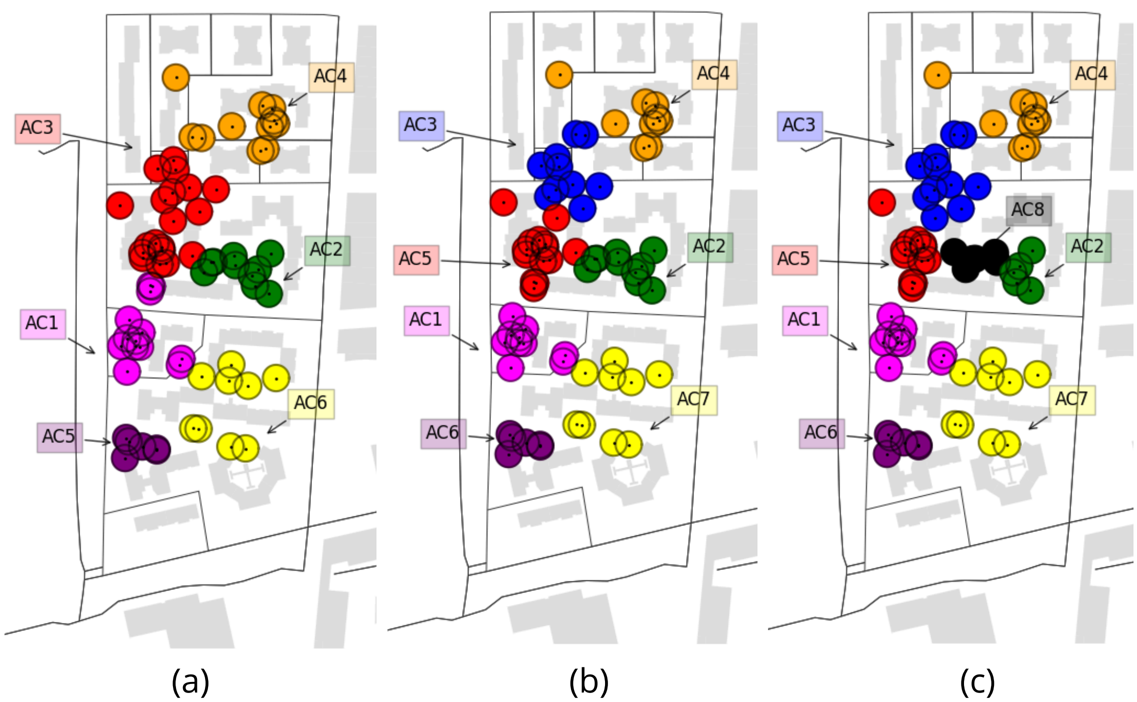


Figure S11: Spatial partitions obtained for different numbers of clusters. Clustering results obtained using the weighted K-means algorithm for (a) $K = 6$, (b) $K = 7$, and (c) $K = 8$. Action stops are colored according to cluster membership. The comparison illustrates the effect of varying the number of clusters on the spatial resolution of the partition.

3 Description of the clusters

We provide a detailed contextualization of each of the seven identified clusters, focusing on the physical characteristics of their surrounding urban environment, the duration of action stops, and the types of activities performed.

3.1 Clusters contextualization

AC1 (magenta). This cluster links two green flowerbed areas encompassing all 10 stops. The actions within this cluster exhibit a playful nature, as detailed in Table S3. The stop durations vary considerably: some last approximately 10–15 minutes, a few are notably brief, and a couple extend to around 30 minutes. Notably, this cluster closely aligns with the P02 area identified in the expert-based mapping (see Fig. 5).

AC2 (green). This cluster encompasses a diverse range of actions and is concentrated in a small intermediate area between buildings (see Fig. 5), featuring a drinking fountain. It is notable for having the longest individual stop duration, approximately 50 minutes, and the second-highest average stop duration, nearly 18 minutes. The shortest stop duration is also relatively long, and the cluster exhibits the largest standard deviation in stop durations, around 14 minutes (cf. Table 3). The cluster is not identified in the expert-based mapping and just vaguely coincide with C03 area identified (see Fig. 5).

AC3 (blue). This cluster shows similar statistical values to those observed in AC2 (cf. Table 3). Particularly, 8 out of the 10 stops of the cluster are longer than 15 minutes. Actions took place mainly inside green flowerbeds, but some occasional action was performed on the sidewalk and pedestrian crossing. Actions are varied across all topologies but with a clear predominance of celebratory activities (see Table S3). As shown in Fig. 5, it follows a very strong overlap with the very prominent C01 area identified in the expert-based mapping.

AC4 (orange). This cluster, comprising 11 actions (9 stops), is situated in an area featuring a playground and the *Plaça de la Font*, the neighborhood's sole square and a popular gathering spot for participants post-activity. The average stop duration here is approximately 9 minutes, among the shortest across all clusters. Actions are relatively brief, exhibiting the lowest standard deviation in stop times, suggesting a high degree of consistency in actions duration. Notably, the playground serves as a hub for playful interactions, including ball games (4 out of 11 actions) and social actions such as talking (2 actions) and dancing (2 actions), as detailed in Table S3. However, despite the presence of the square, only one action occurred centrally within it, possibly due to the hard pavement; the majority took place along its periphery, particularly in the surrounding green flowerbeds. Figure 5 shows a very strong overlap with the very prominent C02 area identified in the expert-based analysis.

AC5 (red). This cluster exhibits the highest density of stops, encompassing 14 stops (20% of the total 68 stops, corresponding to 24 actions) within a relatively compact area. Despite this high density, average stop duration of this cluster is below the overall average (see Table 3). The area features a pedestrian walkway flanked by numerous green flowerbeds, enhancing its appeal for festive mobility and contributing to the high frequency of actions observed. The area is next to AC2 and it is just been partially identified in the expert-based mapping as part of the P02 area (see Fig. 5). Notably, this area is particularly attractive to young people; all seven groups of students from the local school stopped here, with nearly all engaging in two or more actions. Celebratory actions are predominant, accounting for eight of the actions, as detailed in Table S3. Despite the presence of spontaneous and brief interactions, participants also utilized the space for extended periods, engaging in multiple simultaneous activities that emphasized comfort and relaxation.

This is evident at the boundary between this cluster and AC2 (green), where two adjacent stops saw groups performing four and five of the six possible actions, with durations of approximately 40 and 50 minutes, respectively. Interestingly, although both stops are in the same location, the second was classified under the AC2 cluster by the K-means algorithm. This area seems to be an interesting location to enhance the urban rest capabilities of the neighborhood.

AC6 (purple). This cluster encompasses six stops, resulting in a total of 12 actions with diverse characteristics. Despite the occurrence of simultaneous actions, the stop durations are relatively brief, with a few instances slightly exceeding 15 minutes. The cluster has the fewest stops is neatly compacted within a green flowerbed, which separates two building blocks. The cluster aligns with the P02 area identified in the expert-based analysis (see Fig. 5). This region reveals once again the attractiveness of these urban elements (green flowerbeds) when using the public space of the neighborhood.

AC7 (yellow). This cluster, with 9 stops and 11 actions, partially coincides C04 area identified by the expert-based mapping (see Fig. 5). The stop durations are generally brief, with only one exception of a stop exceeding 40 minutes (see Table 3). The cluster area is characterized by a building block in the middle. Both sides integrate diverse urban spaces, including green flowerbeds, crosswalks, and two hard-paved public squares (one in each side). This configuration offers a substantial amount of high-quality public space, facilitating a wide range of actions. The cluster accommodates individuals across all age groups, engaging in various activities, highlighting the area's versatility and appeal.

3.2 Clusters vs Actions

Table S3 displays the number of actions performed in each of the weighted K-means clusters, depending on the topology of the action. There were 6 social actions to do in form of missions [14].

	AC1	AC2	AC3	AC4	AC5	AC6	AC7
J	5	4	3	4	4	1	3
M	2	2	2	2	3	4	0
B	3	3	2	2	4	1	3
X	0	3	1	2	5	3	3
C+CC	3	6	6	1	8	3	2
Total actions	13	18	14	11	24	12	11
Stops	10	10	10	9	14	6	9

Table S3: Number of actions of each type in each of the seven activation clusters. In the left column, the code of each of the actions (J is play, M is lunch, B is dance, X is chat, C is celebration (toast) and CC is celebration (confetti). The last two rows are respectively the total number of actions and the total number of stops in each cluster (which may not coincide as two or more actions could haven been performed at the same stop).

4 Sensitivity analysis of the stop-detection threshold

Stops were detected by analyzing the time difference between consecutive GPS timestamps. Following the procedure described in the associated Data Descriptor [14], we use a threshold of 10 seconds as the minimum time difference to label a record as a stop. This choice was supported by in-field validation tests and reverse engineering of the recording logic of the data-collection app [14]: thresholds below 10 s tend to over-detect short micro-pauses, while substantially larger thresholds progressively merge distinct stops.

To further assess robustness, we performed a sensitivity analysis by varying the stop-detection threshold between 4 and 30 seconds. For each threshold value, we recomputed (i) the total number of detected stops and (ii) the mean stop duration and its standard deviation. Figure S12 summarizes the results. Thresholds below approximately 7 seconds lead to a sharp increase in the number of detected stops, consistent with over-detection of micro-pauses. For larger thresholds, the number of stops decreases smoothly as temporally close pauses are progressively merged, while the mean stop duration increases accordingly.

The selected 10-second threshold lies beyond the over-detection regime observed for thresholds below approximately 7 seconds, and within the region where the variation of the number of detected stops becomes gradual rather than abrupt. This positioning balances over-detection at low thresholds and excessive merging at higher thresholds. This sensitivity analysis complements the in-field validation reported in [14] and supports the robustness of the stop-detection procedure used in this study.

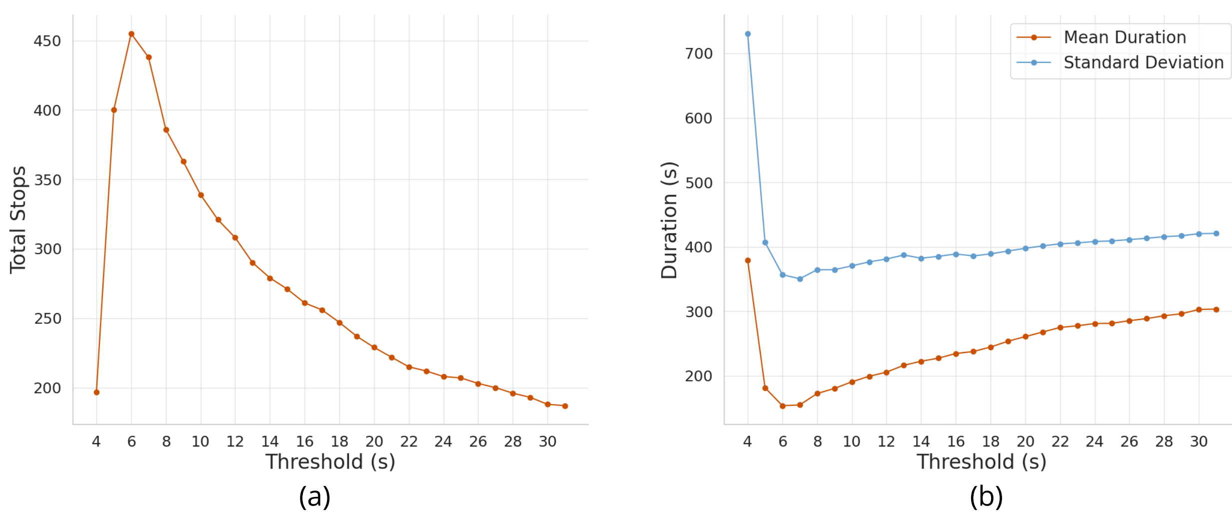


Figure S12: Sensitivity analysis of the stop-detection threshold. (a) Total number of detected stops as a function of the threshold (4–30 s). (b) Mean stop duration and standard deviation as a function of the threshold. Thresholds below ∼7 s over-detect short micro-pauses, while larger thresholds progressively merge distinct stops. The selected 10 s threshold lies within the stable regime.

Supplementary References

[1] Paez, R., Valtchanova, M., Larroya, F. & Perelló, J. Maps as design tools: Space, time and experience. In: Rossetto, T. & Lo Presti, L. The Routledge Handbook of Cartographic Humanities, pp. 172-181 (Routledge, 2024). https://doi.org/10.4324/9781003327578

[2] Valtchanova, M., & Paez, R. Operative Mapping and Collaborative Actions as Design Tools for Critical Socio-spatial Urban Interventions (Repurposing Places, 133. Proceedings of the International Confer-

ence Repurposing Places for Social and Environmental Resilience Editor: Anastasia Karandinou Published 2023)

[3] Hartigan, J. A. & Wong, M. A. A k-means clustering algorithm. *J. R. Stat. Soc., C: Appl. Stat.* 28(1), 100-108 (1979). https://doi.org/10.2307/2346830

[4] Arthur, D., & Vassilvitskii, S. k-means++: The advantages of careful seeding. *Stanford* (2006). https://theory.stanford.edu/~sergei/papers/kMeansPP-soda.pdf

[5] Bholowalia, P. & Kumar, A. EBK-means: A clustering technique based on elbow method and k-means in WSN. *Int. J. Comput. Appl.* 105(9) (2014). https://doi.org/10.5120/18405-9674

[6] Rousseeuw, P.J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* 20, 53-65 (1987). https://doi.org/10.1016/0377-0427(87)90125-7

[7] Tibshirani, R., Walther, G., & Hastie, T. Estimating the number of clusters in a data set via the gap statistic. *J. R. Stat. Soc., B: Stat. Methodol.* 63(2), 411-423 (2001). https://doi.org/10.1111/1467-9868.00293

[8] Ester, M., Kriegel, H. P., Sander, J., & Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*, 226-231 (1996). https://cdn.aaai.org/KDD/1996/KDD96-037.pdf

[9] Lange, T., Roth, V., Braun, M. L., & Buhmann, J. M. Stability-based validation of clustering solutions. *Neural computation* 16(6), 1299–1323 (2004). https://doi.org/10.1162/089976604773717621

[10] Hennig, C. Cluster-wise assessment of cluster stability. *Computational Statistics & Data Analysis* 52(1), 258–271 (2007). https://doi.org/10.1016/j.csda.2006.11.025

[11] Hubert, L., & Arabie, P. Comparing partitions. *Journal of classification* 2(1), 193–218 (1985). https://doi.org/10.1007/BF01908075

[12] Zandbergen, P. A. Accuracy of iPhone locations: A comparison of assisted GPS, WiFi and cellular positioning. *Transactions in GIS* 13, 5–25 (2009). https://doi.org/10.1111/j.1467-9671.2009.01152.x

[13] Monti, S., Tamayo, P., Mesirov, J., & Golub, T. Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Machine learning* 52(1), 91–118 (2003). https://doi.org/10.1023/A:1023949509487

[14] Larroya, F., Paez, R., Valtchanova, M. & Perelló. Explorative pedestrian mobility GPS data from a citizen science experiment in a neighbourhood. *arXiv preprint* arXiv:2410.08672 (2024). https://doi.org/10.48550/arXiv.2410.08672