Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 8, No. 2, April, 2024

Improved Fingerprint-Based Localization Based on Sequential Hybridization of Clustering Algorithms

Abdulmalik Shehu Yaro [1, 2*], Filip Maly [1], Pavel Prazak [1], Karel Malý [1]

[1] *Department of Informatics and Quantitative Methods, Faculty of Informatics and Management, University of Hradec Kralove, 500 03 Hradec Kralove, Czech Republic.*

[2] *Department of Electronics and Telecommunications Engineering, Ahmadu Bello University, Zaria, 810106, Nigeria.*

Abstract

The localization accuracy of a fingerprint-based localization system is dependent on several factors, one of which is the accuracy and efficiency at which the fingerprint database is clustered. Most highly efficient and accurate clustering algorithms have high time-dependent computational complexity (CC), which tends to limit their practical applicability. A technique that has yet to be explored is the sequential hybridization of multiple low-time CC clustering algorithms to produce a single moderate-time CC clustering algorithm with high localization accuracy. As a result, this paper proposes a clustering algorithm with a moderate time CC that is based on the sequential hybridization of the closest access point (CAP) and improved k-means clustering algorithms. The performance of the proposed sequential hybrid clustering algorithm is determined and compared to the modified affinity propagation clustering (m-APC), fuzzy c-mean (FCM), and 2-CAP algorithms presented in earlier research works using four experimentally generated and publicly available fingerprint databases. The performance metrics considered for the comparisons are the position root mean square error (RMSE) and clustering time based on big O notation. The simulation results show that the proposed sequential hybrid clustering algorithm has improved localization accuracy with position RMSEs of about 54%, 77%, and 52%, respectively, higher than those of the m-APC, FCM, and 2-CAP algorithms. In terms of clustering time, it is 99% and 79% faster than the m-APC and FCM algorithms, respectively, but 90% slower than the 2-CAP algorithm. The results have shown that it is possible to develop a clustering algorithm that has a moderate clustering time with very high localization accuracy through sequential hybridization of multiple clustering algorithms that have a low clustering time with poor localization accuracy.

Keywords:

Clustering;

Closest AP;

k-NN;

Sequential Hybrid;

localization Accuracy;

RSS;

k-Means.

Article History:

Received:	20	November	2023
Revised:	14	February	2024
Accepted:	02	March	2024
Published:	01	April	2024

1- Introduction

A fingerprint-based indoor wireless localization system uses position-dependent signal parameters (PDSP) such as the received signal strength (RSS) detected from spatially deployed wireless access points (APs) to determine the location of an indoor radio frequency (RF) transmitter [1]. The fingerprint-based system localization process is divided into two phases: offline and online [2, 3]. The offline phase of the localization process involves the creation of a fingerprint database, also known as a radio map [4–6]. This process involves collecting RF signals at multiple locations known as reference locations (RL), calculating the RSS measurement of each received RF signal, and creating fingerprint vectors [3, 7]. The fingerprint vector consists of all RSS measurements obtained from spatially deployed wireless APs at a single RL. All the fingerprints created are stored in a database, which is known as the fingerprint database. The second phase of the fingerprint-based localization process is known as the online phase. It involves the determination of

* **CONTACT**: abdulmalik.yaro@uhk.cz

DOI: http://dx.doi.org/10.28991/ESJ-2024-08-02-02

© 2024 by the authors. Licensee ESJ, Italy. This is an open access article under the terms and conditions of the Creative Commons Attribution (CC-BY) license (https://creativecommons.org/licenses/by/4.0/).

the real-time location of an RF transmitter using the instantaneously acquired fingerprint measurement by searching through the fingerprint database using a localization matching algorithm such as the k-nearest neighbor (k-NN) [2, 5, 8]. The RL of the fingerprint in the database with the highest degree of similarity to the inputted fingerprint is returned as the real-time position of the RF transmitter.

The localization accuracy of an RSS-based fingerprint system depends on several factors, one of which is the density of the fingerprint database created [2, 9]. The density of the fingerprint database is a function of the number of RLs and their corresponding fingerprint measurements. The higher the density of the fingerprint database, the more RLs are used. In terms of localization accuracy, the higher the fingerprint database density, the higher the localization accuracy. However, this results in the localization matching algorithm taking longer to scan through the fingerprint database, which results in a longer localization time. Fingerprint database clustering has been proposed as a way of achieving high localization accuracy while still utilizing the highly dense fingerprint database [1, 4, 9–11]. As such, the localization accuracy of a clustered fingerprint database is now dependent on the clustering efficiency of the clustering algorithm used. Most of the highly efficient clustering algorithms that will result in high localization accuracy have high computational complexity (CC), resulting in longer clustering times [11]. This limits their practical applicability.

Clustering algorithm modification and fingerprint database pre-processing are the two major techniques used by researchers to reduce clustering time. Clustering algorithm modification involves the modification of the clustering algorithm with the aim of improving its time CC. For instance, the authors in Abed & Abdel-Qader [12] modified the k-means algorithm iteration process by removing fingerprints that are unlikely to change their cluster in subsequent iterations. Another modified version of the k-means algorithm, known as the canopy + k-means algorithm, has also been used to reduce clustering time [13]. The canopy algorithm aids in identifying large fingerprint clusters, which are then refined by the k-means algorithm. In Baldini [14], a fast density-based clustering (DBSCAN) algorithm was presented that is based on a modification of the traditional DBSCAN algorithm using the MapReduce (MR) technique. The MR technique involves parallel processing using a distributed computing cluster. The second technique used by researchers to improve clustering time is fingerprint database pre-processing. This involves reducing the dimensionality of the fingerprint database or removing irrelevant features before clustering. For example, in Chiang et al. [15], principal component analysis (PCA) was used to reduce the size of a fingerprint database, which was then clustered using the adaptive hierarchical clustering algorithm. In Klus et al. [16], fingerprint database size reduction was achieved through the elimination of RSS measurements from redundant wireless APs. Data compression algorithms such as the symbolic aggregate approximation (SAX) algorithm presented in Li et al. [17] and the reduced" alphabet" of value algorithm presented in Sagheer and Yousif [18] have also been used to reduce the size of the fingerprint database. While these techniques can significantly reduce clustering time, they often come with limitations. For instance, fingerprint database pre-processing-based techniques such as dimensionality reduction and compression result in the loss of fingerprint database features, which potentially affects clustering accuracy and subsequently leads to a decrease in localization accuracy. Furthermore, techniques based on algorithm modification sacrifice clustering accuracy for faster execution, which also results in decreased localization accuracy.

A technique that has not yet been fully utilized is the sequential hybridization of several low-time CC clustering algorithms. It involves combining multiple low-time CC clustering algorithms with poor localization accuracy to obtain an improved single algorithm with high localization accuracy and moderate-time CC. As a result, this paper proposes a moderate-time CC clustering algorithm with high localization accuracy based on the sequential hybridization of two popular low-time CC clustering algorithms with poor localization accuracy. This paper makes the following contributions: (a) identifying two highly effective clustering algorithms with low time CC that, when hybridized, yield a moderate time CC clustering algorithm; and (b) developing a sequential hybrid clustering algorithm with moderate time CC and high localization accuracy.

2- Overview of Commonly Used Clustering Algorithms with Low Computational Complexity

The k-means algorithm, fuzzy c-means (FCM) algorithm, affinity propagation clustering (APC) algorithm, DBSCAN algorithm, support vector machines (SVM), Gaussian mixed models (GMM), and neural networks (NN) are five of the most commonly used clustering algorithms that find application in a wide area, including fingerprint database clustering [2, 4, 5, 8, 11, 19]. All these clustering algorithms are very efficient when the optimal initial hyperparameters are used; however, they have different time CC [11]. The k-means algorithm, which is the most commonly used, has the least time CC. This is followed by the FCM algorithm with a moderate-time CC. The GMM, APC, SVM, and DBSCAN algorithms have relatively high time CC. Apart from the five traditional clustering algorithms, there is another type of clustering algorithm that is very effective and has been extensively used in fingerprint database clustering. This algorithm is known as the closest AP (CAP) algorithm [1, 9]. This type of algorithm clusters fingerprints based on the wireless APs closest to each fingerprint. For instance, two fingerprints belong to the same cluster if the wireless AP closest to each fingerprint is the same and is usually identified by the wireless AP with the highest RSS measurement in the fingerprint vector. The CAP algorithm, despite being efficient, has been known to have a very low localization accuracy. Using the big O notation, the time CC of the above-mentioned clustering algorithm can be seen in Table 1 [9, 11].

Table 1. Time CC comparison of some clustering algorithms

Clustering algorithm	Time CC	Symbol and notations	Comment
CAP	$O(d^2)$	*d* is the number of APs	Low
k-means	$O(nkt)$	*n* is the number of RLs *k* is the number of clusters *t* is the number of iterations	Relatively low
FCM	$O(ncdt)$	*n* is the number of RLs *c* is the number of clusters *d* is the number of APs *t* is the number of iterations	Medium
DBSCAN	$O(n^2)$	*n* is the number of RLs	High
APC	$O(n^{2t})$	*n* is the number of RLs *t* is the number of iterations	Very high
GMM	$O(nkd^3)$	*n* is the number of RLs *k* is the number of Gaussian Component *d* is the number of APs	Very high
SVM	$O(n^3)$	*n* is the number of RLs	Very high

As previously stated, this paper intends to sequentially hybridize two low-time CC clustering algorithms to produce a single clustering algorithm with a moderate-time CC that has a very high localization accuracy. Based on the time CC comparison presented in Table 1, the k-means and CAP clustering algorithms have the lowest time CC and will therefore be considered for sequential hybridization. The next subsection describes the clustering methodology for the proposed sequential hybrid clustering algorithm.

3- Proposed Sequential Hybrid Clustering Algorithm Methodology

In this section of the paper, the methodology for the proposed sequential hybrid clustering algorithm is presented. Since the proposed hybrid clustering algorithm is based on the sequential hybridization of the CAP and k-means clustering algorithms, the clustering processes of these two algorithms are first presented. This is followed by the methodology for the sequential hybridization.

3-1- CAP and K-means Algorithms Clustering Methodology

This section of the paper first introduces the clustering methodology for the k-means clustering algorithm, followed by the clustering methodology for the k-means algorithm.

3-1-1- CAP Algorithm Clustering Methodology

The CAP algorithm, as previously stated, is one of two low-time CC clustering algorithms being considered for sequential hybridization. This algorithm is known to have very low time CC; however, the clusters it produces have poor localization accuracy. The CAP algorithm groups fingerprints according to the wireless AP closest to the RL, where each fingerprint is collected [1, 9]. Given a fingerprint vector generated at a given RL, the wireless AP closest to that RL is identified by the wireless AP with the highest RSS value in the given fingerprint vector. The total number of clusters generated by the CAP algorithm depends on the number of wireless APs deployed. For a total of N wireless APs, the CAP algorithm groups the fingerprint into N clusters. Each cluster is given an identification number known as the cluster label, and for the CAP algorithm, the cluster label number is the label of the wireless AP with the highest RSS value amongst the fingerprint clustered. The cluster label for each of the clusters generated by the CAP algorithm is mathematically presented as shown in Equation 1.

$$\mathrm{CL}_i = x_i, \text{ for } 1 \leq i \leq N \tag{1}$$

where "x_i" denotes the labels of the wireless AP with the highest RSS values. In instances where two wireless APs are equidistance from a given RL, the fingerprint corresponding to that RL is assigned to the cluster of the wireless AP with the least cluster label number. A summary of the cluster labelling process of the CAP algorithm given a total of 4 wireless AP each labelled #1, #2, #3 and #4.

Table 2. CAP algorithm cluster labelling methodology

Cluster label and Centroid	Fingerprint cluster description
$CL_1 = 1$	Contain fingerprints, with AP #1 having the highest RSS value.
$CL_2 = 2$	Contain fingerprints, with AP #2 having the highest RSS value.
$CL_3 = 3$	Contain fingerprints, with AP #3 having the highest RSS value.
$CL_4 = 4$	Contain fingerprints, with AP #4 having the highest RSS value.

A summary of the CAP algorithm clustering methodology is presented below, with a methodological flow chart presented in Figure 1.

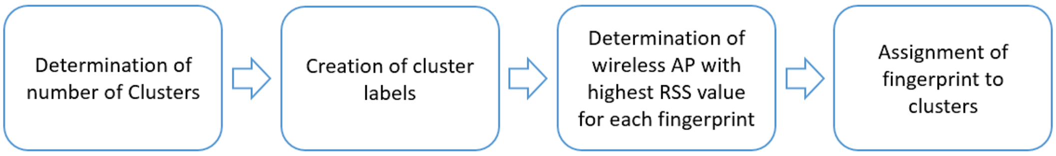

Figure 1. **CAP algorithm clustering methodology**

Step 1: Determine the total number of clusters to be formed.

Step 2: Create the cluster labels using Equation 1 for each cluster.

Step 3: Determine the wireless AP label with the highest RSS value for each fingerprint vector.

Step 4: Assign the fingerprints to the cluster based on the label determined in Step 3.

The CAP algorithm clustering methodology shown in Figure 1 consists of four different stages. The first stage is the cluster number determination stage, in which the total number of clusters to be generated by the CAP algorithm is determined. The second stage is the cluster labelling stage in which each cluster generated in the first stage are labelled using the labelling format presented in Equation 1. The third stage is the determination of the clusters for each fingerprint in the database. This involves identifying the wireless AP label with the highest RSS value in each fingerprint. The final stage of the CAP algorithm clustering method is the cluster assignment stage, which involves assigning fingerprints to the clusters based on the wireless AP label determined in the previous stage.

In the next subsection, the clustering methodology of the second low time CC algorithm identified for sequential hybridization with the CAP algorithm is presented.

3-1-2- K-Means Algorithms Clustering Methodology

The k-means algorithm is the second low-time CC clustering algorithm considered for sequential hybridization with the CAP algorithm. It is the simplest and most robust clustering algorithm that has applications in other areas apart from fingerprint database clustering [1, 2, 5, 11, 20]. The k-means algorithm groups the fingerprints into a predefined number of clusters (k), with each cluster having a fingerprint cluster centroid known as its cluster representative. The fingerprint cluster centroid is obtained by averaging all the fingerprints within that cluster. A summary of the k-means algorithm clustering methodology is shown below, with a graphical representation of the clustering methodology shown in Figure 2.

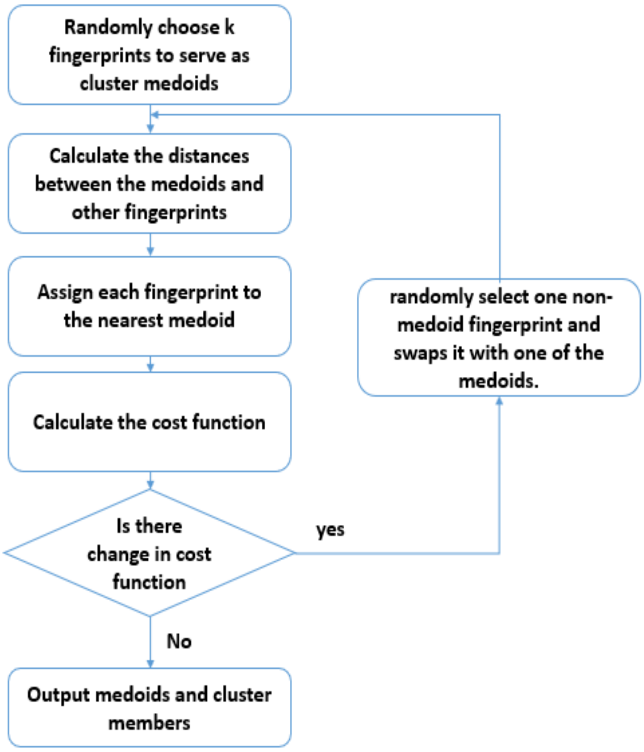

Figure 2. **K-means algorithm clustering methodology**

Step 1: Choose the number of clusters to be formed.

Step 2: Randomly generate and initialize the fingerprint centroids based on the number of clusters in Step 1.

Step 3: Determine the distances between the fingerprint centroids generated in Step 2 and each fingerprint.

Step 4: Assign each fingerprint to the cluster whose fingerprint centroid is closest to it.

Step 5: Recalculate the fingerprint centroids of each cluster by finding the mean of the fingerprints assigned to it.

Step 6: Repeat steps 3 to 5 until there is no change in the fingerprint centroid of each cluster.

The clustering performance of the k-means algorithm is affected by several factors, one of which is the fingerprint cluster centroid initialization process [21, 22]. The conventional k-means algorithm randomly chooses the initial fingerprint cluster centroids, and being an iterative algorithm in which cluster centroids are updated at each iteration, the algorithm will only consider the closest fingerprints to the last centroids when updating them. This can result in the algorithm becoming stuck in local optima, which are suboptimal solutions. K-means++ and farthest point sampling are two techniques commonly used to deterministically initialize the fingerprint cluster centroids. Both techniques are very effective at improving the cluster centroid initializations of the k-means algorithm. K-means++ is more commonly used due to its proven effectiveness and theoretical guarantees [21]. As such, it will be used as the fingerprint centroid initialization process for the k-means algorithm in this paper.

A summary of the k-means++ fingerprint centroid initialization process is shown below [21]:

Step 1: Choose a fingerprint randomly from the database as the first centroid.

Step 2: For each remaining fingerprint, calculate the squared Euclidean distance to the nearest already chosen fingerprint centroid.

Step 3: Choose the next fingerprint centroid from the remaining fingerprints with a probability proportional to the squared distance. Fingerprints that are farther from the already-chosen fingerprint centroids are more likely to be selected as the next fingerprint centroid.

Step 4: Repeat Steps 2 and 3 until all K fingerprint centroids have been selected.

To further improve the fingerprint centroid initialization process of k-means++, the squared Euclidean distance used in Step 2 to determine the nearest fingerprint to the fingerprint centroid is replaced with the inverse weighted distance (IWD) [23]. The combination of k-means++ and IDW allows for better initialization of fingerprint centroids and then refinement of cluster centroids based on the underlying distance relationships in the database using the IDW. This combination leads to more accurate and meaningful clusters, especially when the fingerprint database exhibits spatial patterns or has a strong distance-based structure. In the next subsection, the methodology for the proposed sequential hybrid clustering algorithm is presented.

3-2- Proposed Sequential Hybrid Clustering Algorithm Methodology

In this subsection of the paper, the methodology for the proposed sequential hybrid clustering algorithm is presented. As earlier stated, most authors from previous research works focus on algorithm modifications or use of data pre-processing technique to improve clustering time but this comes at the expense of clustering and localization accuracy. However, in this paper, a completely different approach is used to ensure that clustering time is moderate while maintaining high localization accuracy. The approach presented in this paper is based on the sequential hybridization of low-time CC clustering algorithms with low localization accuracy. As earlier stated, sequential hybridization of clustering algorithms is the process of combining two or more different clustering algorithms in a sequential order to achieve better performance than either algorithm could achieve alone. The goal is to use each algorithm's strengths to offset the other algorithm's weaknesses. The two low-CC clustering algorithms considered for sequential hybridization are the CAP and improved k-means algorithms, which use the clustering methodologies described in Section 3.1.

The order of the clustering sequence for the sequential hybridization method is very important, as it has a significant impact on the overall performance of the resulting clustering process. So, it is important to identify the strengths and weaknesses of each of the two clustering algorithms in order to find the best clustering sequence order. The CAP algorithm is known for being very fast and efficient at clustering fingerprints, as it groups them based on the dominant and closest wireless AP to each fingerprint. By doing so, this eliminates fingerprint outliers and improves the fingerprint database structure. However, it has very poor localization accuracy. The k-means algorithm, on the other hand, is very sensitive to outliers and has poor clustering performance when dealing with complex fingerprint database structures. However, it has good localization accuracy when used on an outlier-free and well-structured fingerprint database.

Based on the strengths and weaknesses comparison of the two clustering algorithms, the optimum clustering sequential order is to use the CAP algorithm first, followed by the k-means algorithm. By so doing, the CAP algorithm eliminates outliers and provides a well-structured fingerprint database for the k-means algorithm to further cluster. A flow chart for the clustering methodology of the proposed sequential hybrid clustering algorithm is shown in Figure 3 below. The clustering methodologies for the CAP and improved k-means algorithms are described in Sections 3.1.1 and 3.1.2, respectively.

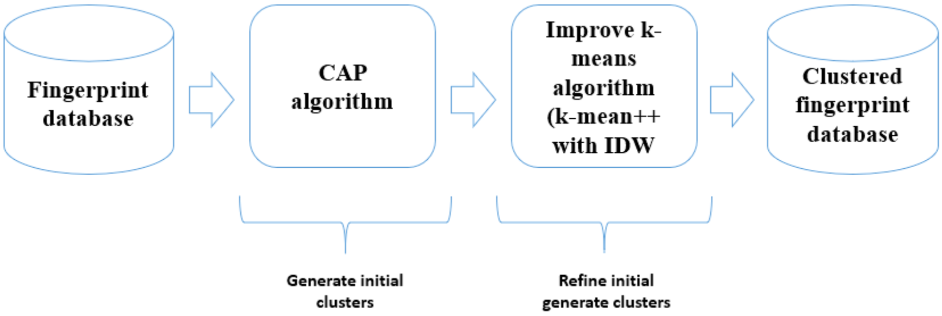

Figure 3. **Clustering methodology for the proposed sequential hybrid clustering algorithm**

A further description of the methodology for how the clusters are generated by the proposed sequential hybrid clustering algorithm using a fingerprint database generated with a total of four wireless APs is shown in Figure 4. It is assumed that the improved k-means algorithm will generate a total of three clusters (k = 3)

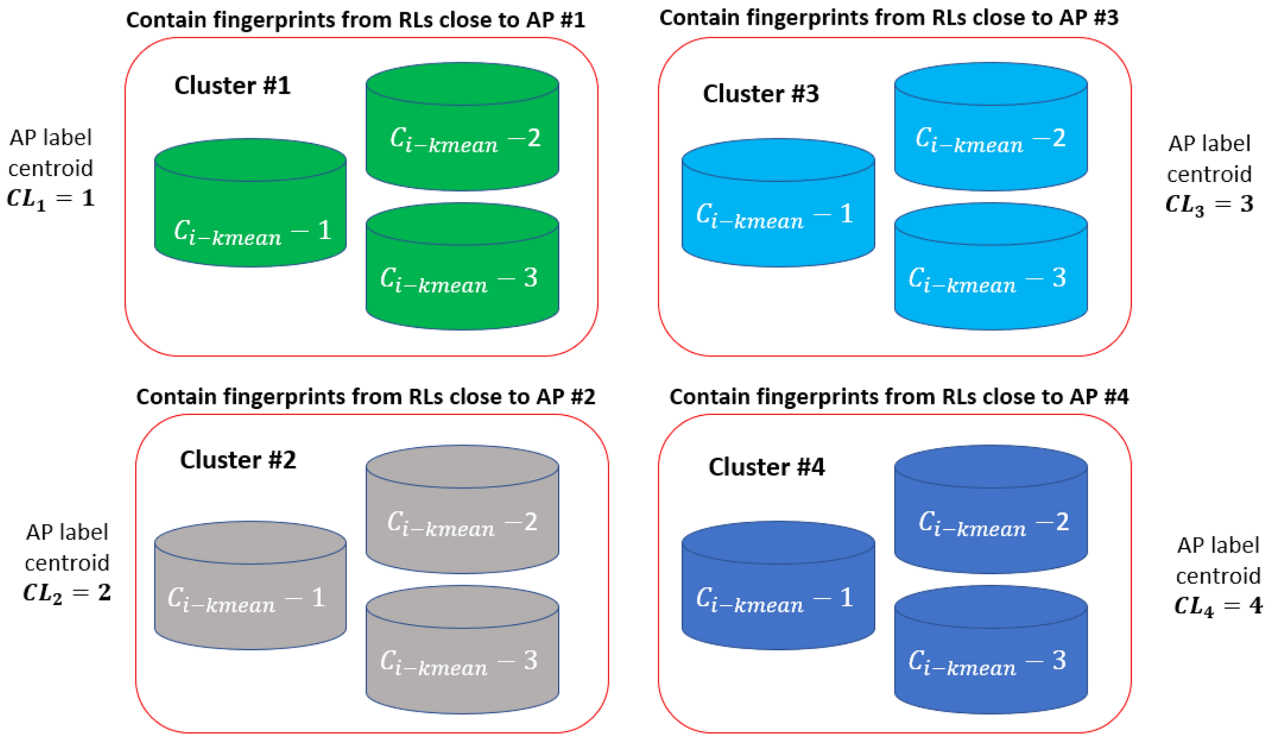

Figure 4. Description of clusters generated by the proposed sequential hybrid clustering algorithm based on 4 wireless APs and k = 3

Figure 4 shows that the CAP algorithm generates the first initial clusters, with the number of initial clusters determined by the number of wireless APs, which in this case is four. Based on Eq. (1), the four initial clusters are assigned the following labels: CL-1, CL-2, CL-3, and CL-4. It is expected that these four initial clusters are free of outliers and well-structured. Each of the four initial fingerprint clusters is then further clustered using the improved k-means algorithm. The improved k-means algorithm is set up to generate three clusters, implying that three clusters are generated within each of the initial clusters. In total, twelve clusters are generated by the proposed sequential hybrid clustering algorithm using a fingerprint database with four wireless APs, with the number of clusters to be generated by the improved k-means algorithm set to three (k = 3).

After the fingerprint database has been clustered using the proposed sequential hybrid clustering algorithm, it is necessary to know how to determine the exact cluster a given fingerprint belongs to in order to extract its RL. This is required for the online phase of the fingerprint-based localization process. Based on what is presented in Figure 4, it can be assumed that two-level clusters are generated. The first-level clusters are generated by the CAP algorithm, while the second-level clusters are generated by the improved k-means algorithm. The first-level cluster to which a given fingerprint belongs can be determined by finding the absolute differences between all the first-level cluster labels and the label of the wireless AP with the highest RSS value of the given fingerprint. The cluster whose cluster label results in a zero absolute difference value is assumed to be the first-level cluster the fingerprint is in. The next step is to determine the second level cluster which is within the first-level cluster already identified. The second-level cluster can be determined by first finding the Euclidean distance between the fingerprint centroid of all the clusters within the first-level cluster and the given fingerprint vector. The cluster with the least Euclidean distance is assumed to contain the given fingerprint. A summary of the methodology for determining the cluster of a given fingerprint based on a fingerprint database clustered by the proposed sequential hybrid clustering algorithm is shown below.

Step 1: Given a fingerprint vector, determine the label of the wireless AP with the highest RSS value and present it as shown in Equation 2.

$$\mathrm{CL}_{input} = x_{input} \tag{2}$$

where x_{input} is the label of the wireless AP with the highest RSS value.

Step 2: Determine the absolute difference between CL_{input} in Equation 2 and first-level cluster labels using Equation 3.

$$\alpha_i = \left|CL_{input} - CL_i\right| \qquad \text{for} \qquad 1 \le i \le N_{cluster} \tag{3}$$

where $N_{cluster}$ is the total number of clusters.

Step 3: The cluster with $\alpha_i = 0$ in Equation 3 is considered to be the first-level cluster.

Step 4: Find the Euclidean distances between all the fingerprint centroids of the subclusters within the first-level cluster and the inputted fingerprint.

Step 5: The subcluster with the least distance from Step 4 is considered to be the second-level cluster where the fingerprint is located.

Following the identification of the subcluster in which the fingerprint is located, the cluster is scanned using the k-NN algorithm to find the RL whose fingerprint matches the inputted fingerprint. The RL obtained is regarded as the estimated location from which the input fingerprint was obtained. Figure 5 shows an overview of the entire localization methodology of a fingerprint-based localization system with the proposed sequential hybrid clustering algorithm. The areas with green highlights are the improvements induced by the proposed sequential hybrid clustering algorithm.

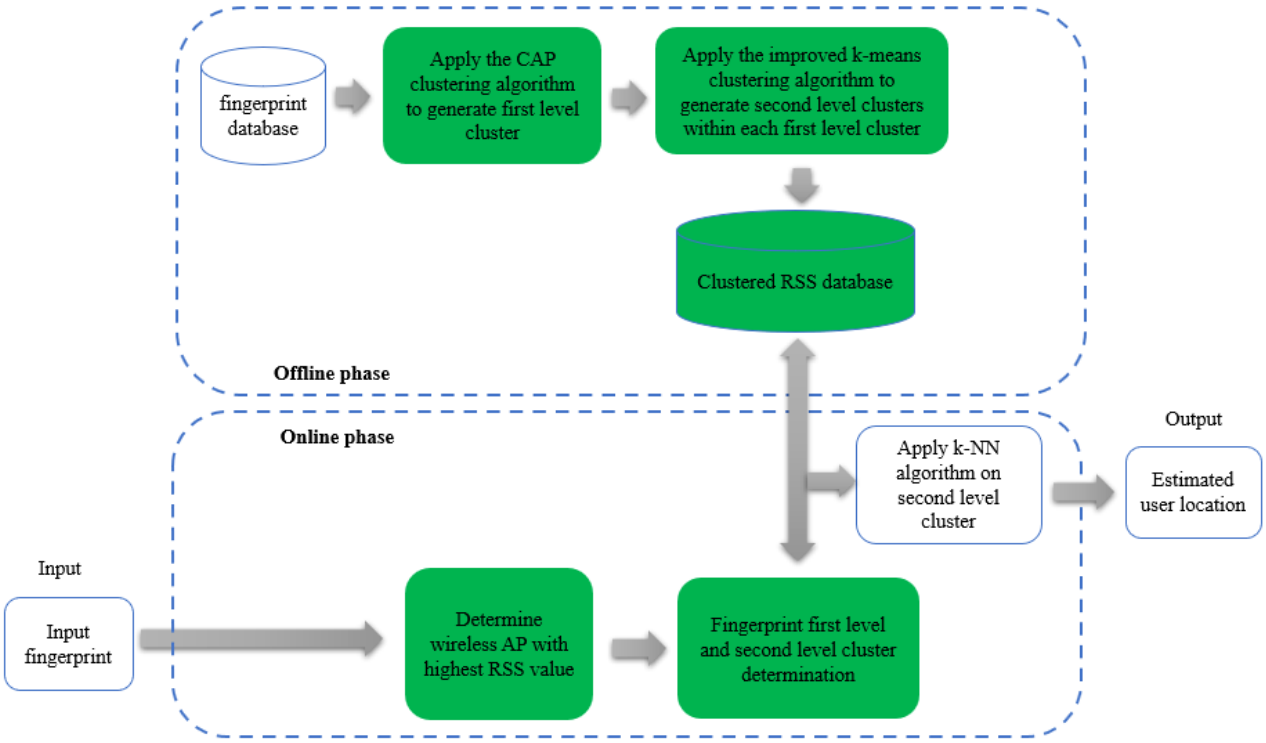

Figure 5. Overview of the localization methodology for the proposed sequential hybrid clustering algorithm

In the following section of this paper, the localization performance of the fingerprint database clustered with the sequential hybrid clustering algorithm is determined and compared to other clustering algorithms presented in previous related research works.

4- Simulation Results and Discussion

This section determines and compares the localization performance of the fingerprint database clustered using the proposed sequential hybrid clustering algorithm with the methodology presented in Section 3.2 to that obtained by other clustering algorithms presented in previous research works. The simulation setup and parameters are presented first, then the localization performance is determined and compared.

4-1- Simulation Setup and Parameters

Four publicly available and experimentally generated RSS-based fingerprint databases are used to evaluate the localization performance of the proposed sequential hybrid clustering algorithms. These four RSS-based fingerprint databases are SEUG_IndoorLoc [24], IIRC_IndoorLoc [25], MSI_IndoorLoc [26], and PIEP_UM_IndoorLoc [27]. The characteristics of the four RSS-based fingerprint databases are shown in Table 3.

Table 3. RSS-based fingerprint database characteristics

Fingerprint Database	Database characteristic			
	Wireless technology	Number of APs	Indoor coverage area	Number of RL
IIRC_IndoorLoc	Zigbee	4	161.12 m^2	194
SEUG_ IndoorLoc	Wi-Fi	3	33 m^2	49
MSI_IndoorLoc	Wi-Fi	11	1000 m^2	4973
PIEP_UM_IndoorLoc	Wi-Fi	8	1000 m^2	1000

The IIRC_IndoorLoc database was generated using a total of four Zigbee-based APs in a room with about 161 m^2 of coverage area and consisted of fingerprint measurements from 194 RLs. The SEUG_IndoorLoc database was generated using three wi-fi-based APs in a room with a coverage area of 33 m^2 and consisted of fingerprints from 49 RLs. The MSI_IndoorLoc and PIEP_UM_IndoorLoc databases are among the databases used in the IPIN 2017 and 2019 conferences. The MSI_IndoorLoc database contains fingerprint measurements from 4973 RLs, and each fingerprint is generated using 11 wi-fi-based APs. The PIEP_UM_IndoorLoc was generated within a total coverage area of 1000 m^2 using 8 wi-fi-based APs and contains fingerprint measurements from 1000 RLs.

Using the four fingerprint databases with characteristics shown in Table 3, the localization performance determination and comparison of the proposed sequential hybrid clustering algorithm are presented in the next subsection.

4-2- Localization Performance Comparison of the Proposed Clustering Algorithm

In this subsection of the paper, the localization performance of the proposed sequential hybrid clustering algorithm is determined using the four fingerprint databases with the characteristics shown in Table 3. At first, its performance is compared to that of the CAP and the improved k-means algorithms. Afterward, its performance is compared to other clustering algorithms presented in previous research works. The position root mean square error (RMSE) and clustering time based on time CC are the metrics used to evaluate localization performance.

4-2-1- Performance Comparison with CAP and Improved K-means Algorithms

This section of the paper presents the localization performance comparison of the proposed sequential hybrid clustering algorithm to that of the CAP and improved k-means algorithms, using position RMSE and time CC as performance metrics. This is to determine if the proposed sequential hybrid clustering algorithm outperforms the CAP and k-means algorithms in terms of localization performance while also having a moderate time CC. Using the four fingerprint databases, the position RMSEs obtained by the proposed sequential hybrid clustering algorithm, the CAP algorithm, and the k-means algorithm are determined and presented in Table 4. The green-highlighted entries in Table 4 indicate the clustering algorithm with the lowest RMSE value.

Table 4. Position RMSE comparison of the proposed sequential clustering algorithm with the CAP and improved k-means algorithms

Fingerprint Database	Position RMSE (m^2)		
	CAP	Improve k-means	Proposed Algorithm
SEUG_ IndoorLoc	0.85	0.90	0.15
IIRC_IndoorLoc	4.51	3.67	1.66
MSI_IndoorLoc	2.86	5.13	0.82
PIEP_UM_IndoorLoc	5.10	6.99	4.89

Based on the position RMSE results presented in Table 4 for all four fingerprint databases considered, the proposed sequential hybrid clustering algorithm has the least position RMSE. In the SEUG_IndoorLoc database, the proposed sequential clustering algorithm has a position RMSE of 0.15 m^2, which is the least, followed by the CAP algorithm with a position RMSE of 0.85 m^2, and then the improved k-means algorithm with a position RMSE of 0.9 m^2. The percentage improvement in localization accuracy achieved by the proposed sequential hybrid clustering over the CAP and k-means algorithms is 82% and 83%, respectively. In the IIRC_IndoorLoc database, the proposed sequential hybrid clustering algorithm has the least position RMSE of about 1.66 m^2, which is about 63% and 55% lower than the position RMSEs obtained by the CAP and improved k-means algorithms, respectively. Also, in the MSI_IndoorLoc database, the proposed sequential hybrid clustering algorithm has the lowest position RMSE of about 0.82 m^2. This is about 71% and 84% lower than the position RMSEs obtained by the CAP and the improved k-means algorithm, respectively. In the last fingerprint database considered, which is the PIEP_UM_IndoorLoc database, the proposed sequential hybrid clustering algorithm also has the least position RMSE of about 4.89 m^2, which is about 4% and 5% higher than the position RMSE obtained by the CAP and improved k-means algorithm. The localization percentage improvement achieved by the proposed sequential clustering algorithm over the CAP and k-means algorithms on the PIEP_UM_IndoorLoc database is not as high as what it achieved in the other three fingerprint databases.

Overall, the proposed sequential hybrid clustering algorithm was able to cluster the SEUG_IndoorLoc, IIRC_IndoorLoc, MSI_IndoorLoc, and PIEP_UM_IndoorLoc databases and achieve an average localization accuracy improvement of at least 82%, 55%, 71%, and 3%, respectively, when compared to the performance of each of the individual clustering algorithms that were hybridized. This improvement in localization performance is expected from the proposed sequential hybrid clustering algorithm. As earlier stated, the sequential hybridization of clustering algorithms combines the strengths of the two hybrid clustering algorithms to obtain a highly efficient clustering algorithm. Having the CAP algorithm generate the initial clusters helps remove fingerprint outliers and generate well-structured fingerprint sub-databases for the improved k-means algorithm to cluster. This helped amplify the clustering accuracy of the improved k-means algorithm, which subsequently resulted in the improved localization accuracy achieved by the proposed sequential hybrid clustering algorithm. The results obtained on the SEUG_IndoorLoc, IIRC_IndoorLoc, and MSI_IndoorLoc databases clearly demonstrate the proposed sequential hybrid clustering algorithm's significant improvement in localization accuracy over the two algorithms hybridized.

Extending the result analysis and discussion to the time-CC comparison, the proposed sequential hybrid clustering algorithm has a two-level clustering methodology. The CAP algorithm does the clustering at the first level and has a time CC of $O(d^2)$ from Table 1. The improved k-means algorithm does the second-level clustering, and from Table 1, it has a time CC of $O(nkt)$. The second-level clustering takes place within each of the first-level clusters generated but will be executed in parallel; as such, the overall time CC for the second-level clustering is the same as the time CC for the improved k-means algorithm, which is $O(nkt)$. The total time CC for the proposed sequential hybrid clustering algorithm, considering the time CCs for the first and second levels of clustering, is obtained as $O(d^2 + nkt)$. Using the fingerprint database characteristics listed in Table 3, the time CCs for the CAP algorithm, the improved k-means algorithm, and the proposed sequential hybrid clustering algorithm are determined and shown in Table 5.

Table 5. **Time CC comparison for CAP algorithm, improve k-means algorithm and the propose algorithm**

Fingerprint Database	Time CC (sec)		
	Improved k-means	CAP	Proposed
IIRC_IndoorLoc	582	16	598
SEUG_IndoorLoc	147	9	156
MSI_IndoorLoc	14919	121	15040
PIEP_UM_IndoorLoc	3000	64	3054

From the time CC results in Table 5, the CAP algorithm has the least time CC in all four fingerprint databases, followed by the improved k-means. The proposed sequential hybrid clustering algorithm has the highest CC. Such an increase in time CC by the proposed sequential hybrid clustering algorithm is expected. This is because the proposed sequential hybrid clustering algorithm performs clustering in a sequential order, which means one at a time in a linear fashion. The overall execution time of a sequential-order process is the sum of the execution times of each individual process. What is important is that the time CC of the proposed sequential hybrid clustering algorithm should not be significantly higher than the time CC of the slowest of the two clustering algorithms hybridized. The time CC for the proposed sequential hybrid clustering algorithm is on average 5% and 95% higher than the time CC for the improved k-means and CAP algorithms, respectively, for all four fingerprint databases. The improve k-means algorithm has the slowest time CC, and the proposed sequential hybrid clustering algorithm is on average 5% slower than the improve k-means algorithm. When compared to the degree of improvement in localization accuracy, the proposed sequential hybrid clustering algorithm's 5% increase in clustering time is insignificant.

In summary, based on the four RSS-based fingerprint databases considered, the proposed sequential hybrid clustering algorithm has a localization accuracy that is significantly higher than the individual clustering algorithms hybridized. It has a time CC that is slightly higher than the time CC of the slowest clustering algorithms hybridized, which can be considered insignificant in comparison to the level of localization accuracy achieved. The localization and clustering time performances of the proposed sequential hybrid clustering algorithm are compared to those of other clustering algorithms using the four fingerprint databases, with the characteristics shown in Table 3 in the following section.

4-2-2- Localization Comparison with other Clustering Algorithms Present in Related Works

In this section of the paper, the localization and clustering time performances of the proposed sequential hybrid clustering algorithm are compared to three other clustering algorithms presented in previous studies. The three clustering algorithms considered are the modified APC (m-APC) algorithm presented in Shang and Wang [4], the FCM algorithm presented in Wu et al. [28], and the 2-CAP algorithm presented in Yaro et al. [1]. The k-NN algorithm is considered for the localization with k = 3. Using the four RSS-based fingerprint databases with characteristics shown in Table 3, the position RMSEs obtained by the m-APC, FCM, 2-CAP, and the proposed algorithm are determined and presented in Table 6. The green-highlighted entries in Table 6 indicate the clustering algorithm with the lowest position RMSE.

Table 6. Position RMSEs obtained by the proposed clustering algorithm, the m-APC, FCM, and 2-CAP algorithms

Fingerprint Database	Position RMSE (m^2)			
	m-APC	FCM	2-CAP	Proposed algorithm
SEUG_ IndoorLoc	0.92	0.83	0.75	0.15
IIRC_IndoorLoc	2.86	4.51	2.75	1.66
MSI_IndoorLoc	1.96	4.91	2.62	0.82
PIEP_UM_IndoorLoc	4.43	15.00	3.81	2.98

According to the position RMSE values presented in Table 6, the proposed sequential hybrid clustering algorithm has the lowest position RMSE in all four fingerprint databases. In the SEUG_IndoorLoc database, the proposed sequential hybrid clustering algorithm has the least position RMSE of about 0.15 m^2, which is followed by the 2-CAP algorithm with a position RSME of about 0.75 m^2. Next is the FCM algorithm with a position RMSE of about 0.83 m^2, with the m-APC algorithm having the highest position RMSE of about 0.92 m^2. The localization percentage improvements achieved by the proposed sequential hybrid clustering algorithm over the m-APC, FCM, and 2-CAP algorithms are 84%, 82%, and 80%, respectively. In the IIRC_IndoorLoc database, the proposed sequential hybrid clustering algorithm also has the lowest position RMSE of about 1.66 m^2, which is about 42%, 63%, and 40% lower than the position RMSE obtained by the m-APC, FCM, and 2-CAP algorithms, respectively. Likewise, in the MSI_IndoorLoc database, the proposed sequential hybrid clustering algorithm has the lowest position RMSE of about 0.82 m^2, which resulted in a localization performance improvement of about 58%, 83%, and 69% when compared to the m-APC, FCM, and 2-CAP algorithms, respectively. In the PIEP_UM_IndoorLoc database, the proposed sequential hybrid clustering algorithm has the lowest position RMSE of about 2.98 m^2, with localization performance improvements of about 32%, 80%, and 21% when compared to the m-APC, FCM, and 2-CAP algorithms, respectively.

Overall, based on the earlier result discussions, all databases clustered using the proposed sequential hybrid clustering algorithm resulted in improved localization accuracy. It achieved at least 80%, 40%, 58%, and 21% localization accuracy improvements in the SEUG_IndoorLoc, IIRC_IndoorLoc, MSI_IndoorLoc, and PIEP_UM_IndoorLoc databases, respectively, in comparison to the FCM, 2-CAP, and m-APC algorithms. This has demonstrated the superiority of the proposed sequential hybrid clustering algorithm. It also shows that by sequentially hybridizing multiple low-time CC clustering algorithms with poor localization accuracy, a clustering algorithm with a significant high localization accuracy can be created. However, there is a need to verify the time CC of the proposed sequential hybrid clustering algorithm to see if it is lower or moderate in comparison to the time CC of the FCM, 2-CAP, and m-APC algorithms. Table 7 shows the time CC comparison of the proposed sequential hybrid clustering algorithm to the time CC of the FCM, 2-CAP, and m-APC algorithms for each of the fingerprint databases considered.

From the time CC comparison in Table 7, the 2-CAP algorithm has the fastest time CC (lowest clustering time), which is followed by the proposed sequential clustering algorithm. The next slowest clustering algorithm is the FCM algorithm, while the APC algorithm has the worst time CC (takes the longest to cluster). Generally speaking, the 2-CAP algorithm, which is a modified version of the CAP algorithm, has a very fast clustering time; however, as earlier mentioned, the CAP-based algorithm has the worst localization algorithm. The 2-CAP algorithm is on average 90% faster in clustering than the proposed sequential hybrid clustering algorithm; however, the proposed sequential hybrid clustering algorithm has an average localization accuracy improvement of about 52% across all four fingerprint databases considered in comparison to the 2-CAP algorithm. As for the m-APC and FCM algorithms in comparison, the proposed sequential hybrid clustering algorithm is much faster. The proposed sequential hybrid clustering algorithm has a clustering time that is on average 99% and 79% lower than the clustering times of the m-APC and FCM algorithms, respectively, across all four fingerprint databases considered.

Table 7. Time CC comparison of the proposed sequential hybrid clustering algorithm for one iteration cycle (t = 1).

Fingerprint Database	Time CC (sec)			
	2-CAP	**APC**	**FCM**	**Proposed**
IIRC_IndoorLoc	48	37636	2328	648
SEUG_ IndoorLoc	18	2401	442	156
MSI_IndoorLoc	1210	24730729	164120	15040
PIEP_UM_IndoorLoc	448	1000000	24000	3054

In summary, based on position RMSE comparisons, the proposed sequential hybrid clustering algorithm outperformed the m-APC, FCM, and 2-CAP algorithms by 54%, 77%, and 52%, respectively, across all four fingerprint databases. In terms of time CC comparison, it has a clustering time that is approximately 99% and 79% faster than that of the m-APC and FCM algorithms, respectively, but it is 90% slower than the 2-CAP algorithm. The result analysis based on the four fingerprint databases considered has proved the superiority in terms of localization performance of the proposed sequential hybrid clustering algorithm. This has also proved that it is possible to develop a clustering algorithm that has a moderate clustering time with very high localization accuracy through a sequential hybridization of multiple clustering algorithms that have a low clustering time with poor localization accuracy.

5- Conclusion

This paper explored the possibility of creating a clustering algorithm with high localization accuracy and moderate clustering time by sequentially hybridizing multiple clustering algorithms with low clustering time and poor localization accuracy. The sequential hybridization of clustering algorithms combines the advantages of the two hybrid clustering algorithms to produce a highly efficient clustering algorithm. The CAP and improved k-means clustering algorithms are being considered for hybridization. The CAP algorithm has a very short clustering time but very poor localization accuracy, whereas the improved k-means algorithm is known to be very efficient with good localization accuracy only when used on an outlier-free and well-structured fingerprint database. For the proposed sequential hybrid clustering algorithm, the CAP algorithm is used to generate the initial clusters, which will be outlier-free and well-structured for the improved k-means algorithm to further cluster, resulting in improved clustering accuracy. The performance of the proposed sequential hybrid clustering algorithm is determined using four experimentally generated and publicly available fingerprint databases. Position RMSE and clustering time based on big O notation are used as the localization performance metrics.

The results show that the proposed sequential hybrid clustering algorithm has an improved localization accuracy of at least 82%, 55%, 71%, and 3% on the SEUG_IndoorLoc, IIRC_IndoorLoc, MSI_IndoorLoc, and PIEP_UM_IndoorLoc databases in comparison to each of the two clustering algorithms hybridized. The clustering time for the proposed sequential hybrid clustering algorithm is about 5% higher than the clustering time of the slowest of the two clustering algorithms hybridized. This is expected and considered to be insignificant in comparison to the level of localization accuracy improvement achieved. The proposed sequential hybrid algorithm is further compared to mAPC, FCM, and 2-CAP algorithms, which were present in previous research work. The localization accuracy improvements achieved by the proposed sequential hybrid clustering algorithm are on average 54%, 77%, and 52% higher than those of the mAPC, FCM, and 2-CAP algorithms on all four fingerprint databases considered. It is about 99% and 79% faster in clustering than the m-APC and FCM algorithms, but 90% slower than the 2-CAP algorithm. The findings of this paper demonstrate that sequential hybridization is an effective method for developing a clustering algorithm with a short to moderate clustering time and high localization accuracy. This method is superior to the fingerprint database pre-processing and algorithm modification methods currently employed by other researchers, as these methods are known to trade off localization accuracy for clustering time reduction.

6- Declarations

6-1- Author Contributions

Conceptualization, A.S.Y. and F.M.; methodology, A.S.Y. and K.M.; software, F.M.; validation, A.S.Y.; formal analysis, A.S.Y.; investigation, A.S.Y.; resources, F.M.; data curation, A.S.Y. and K.M.; writing—original draft preparation, A.S.Y.; writing—review and editing, A.S.Y., K.M., and F.M.; visualization, A.S.Y.; supervision, P.P. and F.M.; project administration, P.P. and F.M.; funding acquisition, F.M. All authors have read and agreed to the published version of the manuscript.

6-2- Data Availability Statement

Data sharing is not applicable to this article.

6-3- Funding

This research is supported by the SPEV 2023 project run at the Faculty of Informatics and Management, University of Hradec Kralove, Czech Republic.

6-4- Acknowledgements

The authors would like to thank the Faculty of Informatics and Management (FIM) SPEV Project 2023 for funding this research. The authors would also like to thank Ing. Daniel Schmidt for his valuable help during the data collection and preparation stages of this research work. His dedication and unwavering support greatly enhanced the quality and depth of this research.

6-5- Institutional Review Board Statement

Not applicable.

6-6- Informed Consent Statement

Not applicable.

6-7- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

7- References

[1] Yaro, A. S., Malý, F., & Malý, K. (2023). A Two-Nearest Wireless Access Point-Based Fingerprint Clustering Algorithm for Improved Indoor Wireless Localization. Emerging Science Journal, 7(5), 1762–1770. doi:10.28991/ESJ-2023-07-05-019.

[2] Yong, Y. F., Tan, C. K., Tan, I. K., & Tan, S. W. (2023). C-VoNNI: a precise fingerprint construction for indoor positioning systems using natural neighbor methods with clustering-based Voronoi diagrams. The Journal of Supercomputing, 1-28. doi:10.1007/s11227-023-05855-3.

[3] Yaro, A. S., Maly, F., & Maly, K. (2023). Improved Indoor Localization Performance Using a Modified Affinity Propagation Clustering Algorithm with Context Similarity Coefficient. IEEE Access, 11, 57341–57348. doi:10.1109/ACCESS.2023.3283592.

[4] Shang, S., & Wang, L. (2022). Overview of Wi-Fi fingerprinting-based indoor positioning. IET Communications, 16(7), 725–733. doi:10.1049/cmu2.12386.

[5] Al Mamun, M. A., Anaya, D. V., & Yuce, M. R. (2021). FaStER: Fast, Stable, Expendable and Reliable Radio Map for Indoor Localization. 2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL), Kailua-Kona, United States. doi:10.1109/inertial51137.2021.9430464.

[6] Khoo, H. W., Ng, Y. H., & Tan, C. K. (2022). Enhanced Radio Map Interpolation Methods Based on Dimensionality Reduction and Clustering. Electronics, 11(16), 2581. doi:10.3390/electronics11162581.

[7] Yaro, A. S., Maly, F., & Prazak, P. (2023). A Survey of the Performance-Limiting Factors of a 2-Dimensional RSS Fingerprinting-Based Indoor Wireless Localization System. Sensors, 23(5), 2545. doi:10.3390/s23052545.

[8] Singh, N., Choe, S., & Punmiya, R. (2021). Machine Learning Based Indoor Localization Using Wi-Fi RSSI Fingerprints: An Overview. IEEE Access, 9, 127150–127174. doi:10.1109/ACCESS.2021.3111083.

[9] Quezada-Gaibor, D., Torres-Sospedra, J., Nurmi, J., Koucheryavy, Y., & Huerta, J. (2021). Lightweight Wi-Fi Fingerprinting with a Novel RSS Clustering Algorithm. 2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Lloret de Mar, Spain. doi:10.1109/ipin51156.2021.9662612.

[10] Ezhumalai, B., Song, M., & Park, K. (2021). An efficient indoor positioning method based on Wi-Fi RSS fingerprint and classification algorithm. Sensors, 21(10), 3418. doi:10.3390/s21103418.

[11] Xu, D., & Tian, Y. (2015). A Comprehensive Survey of Clustering Algorithms. Annals of Data Science, 2(2), 165–193. doi:10.1007/s40745-015-0040-1.

[12] Abed, A., & Abdel-Qader, I. (2019). RSS-fingerprint dimensionality reduction for multiple service set identifier-based indoor positioning systems. Applied Sciences (Switzerland), 9(15), 3137. doi:10.3390/app9153137.

[13] Bakhthemmat, A., & Izadi, M. (2020). Decreasing the execution time of reducers by revising clustering based on the futuristic greedy approach. Journal of Big Data, 7(1), 6. doi:10.1186/s40537-019-0279-z.

[14] Baldini, G., Giuliani, R., Steri, G., Sanchez, I., & Gentile, C. (2017). The application of the Symbolic Aggregate Approximation algorithm (SAX) to radio frequency fingerprinting of IoT devices. 2017 IEEE Symposium on Communications and Vehicular Technology (SCVT), Leuven, Belgium. doi:10.1109/scvt.2017.8240312.

[15] Chiang, M. C., Tsai, C. W., & Yang, C. S. (2011). A time-efficient pattern reduction algorithm for k-means clustering. Information Sciences, 181(4), 716–731. doi:10.1016/j.ins.2010.10.008.

[16] Klus, L., Quezada-Gaibor, D., Torres-Sospedra, J., Lohan, E. S., Granell, C., & Nurmi, J. (2020). RSS Fingerprinting Dataset Size Reduction Using Feature-Wise Adaptive k-Means Clustering. 12th International Congress on Ultra-Modern Telecommunications and Control Systems and Workshops (ICUMT), Brno, Czech. doi:10.1109/icumt51630.2020.9222458.

[17] Li, A., Fu, J., Shen, H., & Sun, S. (2021). A Cluster-Principal-Component-Analysis-Based Indoor Positioning Algorithm. IEEE Internet of Things Journal, 8(1), 187–196. doi:10.1109/JIOT.2020.3001383.

[18] Sagheer, N. S., & Yousif, S. A. (2021). Canopy with k-means clustering algorithm for big data analytics. AIP Conference Proceedings. doi:10.1063/5.0042398.

[19] Liu, S., DE Lacerda, R., & Fiorina, J. (2021). WKNN indoor Wi-Fi localization method using k-means clustering based radio mapping. 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Helsinki, Finland. doi:10.1109/vtc2021-spring51267.2021.9448961.

[20] Zhao, H. (2022). Design and Implementation of an Improved K-Means Clustering Algorithm. Mobile Information Systems, 2022, 1–10. doi:10.1155/2022/6041484.

[21] Arthur, D., & Vassilvitskii, S. (2007). K-means++: The advantages of careful seeding. Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, 7-9 January, 2007, New Orleans, United States.

[22] Celebi, M. E., Kingravi, H. A., & Vela, P. A. (2013). A comparative study of efficient initialization methods for the k-means clustering algorithm. Expert Systems with Applications, 40(1), 200–210. doi:10.1016/j.eswa.2012.07.021.

[23] Bi, J., Huang, L., Cao, H., Yao, G., Sang, W., Zhen, J., & Liu, Y. (2021). Improved indoor fingerprinting localization method using clustering algorithm and dynamic compensation. ISPRS International Journal of Geo-Information, 10(9), 613. doi:10.3390/ijgi10090613.

[24] Sadowski, S., Spachos, P., & Plataniotis, K. N. (2020). Memoryless Techniques and Wireless Technologies for Indoor Localization with the Internet of Things. IEEE Internet of Things Journal, 7(11), 10996–11005. doi:10.1109/JIOT.2020.2992651.

[25] Alhmiedat, T. (2023). Fingerprint-Based Localization Approach for WSN Using Machine Learning Models. Applied Sciences, 13(5), 3037. doi:10.3390/app13053037.

[26] Moreira, A., Silva, I., Meneses, F., Nicolau, M. J., Pendao, C., & Torres-Sospedra, J. (2017). Multiple simultaneous Wi-Fi measurements in fingerprinting indoor positioning. 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sapporo, Japan. doi:10.1109/ipin.2017.8115914.

[27] Torres-Sospedra, J., Moreira, A., Mendoza-Silva, G. M., Joao Nicolau, M., Matey-Sanz, M., Silva, I., Huerta, J., & Pendao, C. (2019). Exploiting Different Combinations of Complementary Sensor's data for Fingerprint-based Indoor Positioning in Industrial Environments. 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Pisa, Italy. doi:10.1109/ipin.2019.8911758.

[28] Wu, Y., Jia, Z., Dai, Y., & Wang, W. (2022). Research on Wi-Fi Location Fingerprint Positioning Algorithm Based on DPC-FCM Clustering. Wireless Communications and Mobile Computing, 2022, 1–7. doi:10.1155/2022/8347084.