



A Two-Nearest Wireless Access Point-Based Fingerprint Clustering Algorithm for Improved Indoor Wireless Localization

Abdulmalik Shehu Yaro^{1, 2*} , Filip Malý¹ , Karel Malý¹ 

¹ *Department of Informatics and Quantitative Methods, Faculty of informatics and management, University of Hradec Kralove, Hradec Kralove, 500 03, Czech Republic.*

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

Abstract

Fingerprint database clustering is one of the methods used to reduce localization time and improve localization accuracy in a fingerprint-based localization system. However, optimal selection of initial hyperparameters, higher computation complexity, and interpretation difficulty are among the performance-limiting factors of these clustering algorithms. This paper aims to improve localization time and accuracy by proposing a clustering algorithm that is extremely efficient and accurate at clustering fingerprint databases without requiring the selection of optimal initial hyperparameters, is computationally light, and is easily interpreted. The two closest wireless access points (APs) to the reference location where the fingerprint is generated, as well as the labels of the two APs in vector form, are used by the proposed algorithm to cluster fingerprints. The simulation result shows that the proposed clustering algorithm has a localization time that is at least 45% faster and a localization accuracy that is at least 25% higher than the k -means, fuzzy c -means, and lightweight maximum received signal strength clustering algorithms. The findings of this paper further demonstrate the real-time applicability of the proposed clustering algorithm in the context of indoor wireless localization, as low localization time and higher localization accuracy are the main objectives of any localization system.

Keywords:

Clustering;
C-means; K-means;
RSS;
Fingerprinting;
Localization; Position Error.

Article History:

Received:	14	May	2023
Revised:	11	September	2023
Accepted:	19	September	2023
Published:	01	October	2023

1- Introduction

Indoor wireless localization is the process of determining an indoor user's (IU) instantaneous location with a localization algorithm using position-dependent signal parameters (PDSPs) derived from the IU's received radio frequency (RF) emission [1, 2]. Several PDSPs can be derived from an IU's detected RF emission, including time of arrival (TOA), channel state information (CSI), and received signal strength (RSS); however, this paper focuses on the use of RSS for indoor localization. The choice of RSS is the result of its many benefits, which range from being simple and inexpensive to its robustness to interference and scalability [1, 3]. The two most widely used localization algorithms for locating IUs using RSS measurements are fingerprinting and trilateration. However, because of its accuracy, flexibility, scalability, and robustness to multipath effects and non-line-of-sight (NLOS) propagation, the fingerprinting localization algorithm is considered superior to trilateration and will be considered in this paper [3, 4].

The RSS-based fingerprint system's localization process is divided into two phases, which are called the online phase and the offline phase [1]. The acquisition of signals from several spatially distributed Wi-Fi or Bluetooth (BLE) access points (APs) is the first step in the offline phase of the localization process. The determination of the RSS values for each signal received comes next, and the creation of a fingerprint database comes last [1]. The fingerprint database, also

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

DOI: <http://dx.doi.org/10.28991/ESJ-2023-07-05-019>

© 2023 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/>).

known as the "RSS radio map," is a database that has RSS measurement vectors known as fingerprints mapped to the reference location (RL) from which they were obtained. A fingerprint consists of RSS measurements obtained from multiple Wi-Fi or BLE APs at the same RL. There are several methods [1, 5] used in the generation of the fingerprint database, and the site survey method [1] is the most used. The generation of the fingerprint database is not the focus of this paper; hence, it is assumed that it has been generated using any of the available methods present in the literature.

The online phase of the localization process follows the generation of the fingerprint database. In this phase, the IU's real-time location is determined by searching the fingerprint database for an RL whose fingerprint matches the IU's real-time acquired fingerprint. Several matching algorithms, including k -nearest neighbour (k -NN), support vector machine (SVM), Gaussian mixture model (GMM), and deep learning-based feature extraction methods, have been used to search the fingerprint database [1, 3, 6]. Due to its ease of use, high accuracy, and resistance to noise, the k -NN algorithm is the most common localization-matching algorithm [3]. As a result, it will be used in this paper.

The performance of the RSS-based fingerprint indoor wireless localization system depends on several factors, one of which is the density of the fingerprint database. This is a function of the number of fingerprints and wireless AP used. The density of the fingerprint database has an impact on the localization time and accuracy of the system. The larger the fingerprint database, the higher the localization accuracy, but the longer it takes to determine the location of an IU, and vice versa. To solve this localization time and accuracy trade-off, fingerprint database clustering algorithms are used [7]. However, the performance of these fingerprint clustering algorithms is constrained by a number of factors, including the choice of the best initial hyperparameters, such as cluster number and cluster centroids, as well as the type of fingerprint similarity measure employed and being extremely computationally intensive [7–9]. Therefore, this paper proposes a clustering algorithm that is computationally light and does not necessitate the selection of optimal initial hyperparameters in order to increase localization accuracy and reduce localization time. As a result, the contribution of this paper is as follows: (a) the development of a computationally light clustering algorithm that groups fingerprints into clusters using the two APs that are closest to the RL at which the fingerprint is obtained, and (b) the development and use of AP label-based vectors as cluster centroids to enable the identification of the cluster in which a fingerprint is located.

The remainder of the paper is organised as follows: Section 2 provides a summary of the review related to fingerprint database clustering algorithms. Section 3 describes the proposed clustering algorithm, while Section 4 contains the simulation results and conclusion. Section 5 contains the conclusion and recommendations for future work.

2- Review of Related Works on Fingerprint Database Clustering Algorithms

A number of studies have proposed and used various clustering algorithms with the objective of reducing localization time and improving localization accuracy [7–13]. A clustering algorithm based on the density-based spatial clustering (DBSCAN) algorithm was proposed by Marakkalage et al. [10] but it has been modified to be more noise-resistant. The DBSCAN was modified by replacing the density-based fingerprint similarity measure with the Cosine fingerprint similarity measure. Despite demonstrating improved localization accuracy, the algorithm is computationally intensive, very difficult to understand, and sensitive to the selection of initial hyperparameters, which are the minimum number of fingerprints (MinPts) and the radius of the neighbourhood (Eps) around each fingerprint. In Quezada-Gaibor et al. [8] and Ezhumalai et al. [11], two different clustering algorithms are proposed that do not require the selection of an initial hyperparameter. The algorithm in Ezhumalai et al. [11] uses the strongest access point (SAP) to cluster fingerprints, with SAP label distance as a measure of fingerprint similarity. If the distance between the SAP labels of two or more fingerprints is equal to zero, they are said to belong to the same cluster. The distribution of APs in the indoor environment has a significant impact on this algorithm. The clustering algorithm in Quezada-Gaibor et al. [8] is based on the maximum RSS value, and two or more fingerprints belong to the same cluster if their maximum RSS comes from the same AP. The clustering algorithm produced a very fast localization time but a very poor localization accuracy.

Other clustering algorithms, such as k -means, fuzzy c -means (FCM), and affinity propagation clustering (APC), as well as their modifications, have also been used to cluster fingerprint databases [13–21]. Zhao [13] and Klus et al. [21] clustered an RSS-based fingerprint database using the k -means clustering algorithm. Although the k -means clustering algorithm is simple to use and effective on small fingerprint databases, its performance is constrained by the number of clusters employed and is also highly dependent on the initial cluster centroids chosen. The FCM algorithm used in Sun et al. [14] and its modified version in Wu et al. [16] share the same performance limitations as the k -means, which include choosing the ideal number of clusters to be generated. A clustering algorithm that does not require prior selection of the number of clusters is the APC algorithm [15, 18, 22]. However, the APC algorithm and its modifications have other limitations, such as high computational intensity, being very difficult to interpret, and not being suitable for all fingerprint databases. A summary of all earlier fingerprint database clustering algorithms and their limitations is presented in Table 1.

According to Table 1, each clustering algorithm has one or more limitations. These limitations could be the choice of the optimum initial hyperparameter, which could be the radius of the neighbourhood, the number of clusters, the cluster centroids, the minimum number of fingerprints per cluster, or sensitivity to the distribution of APs in the environment.

High computational demand, difficult interpretation, or very poor localization accuracy are some other limitations. As a result, a clustering algorithm that is not dependent on choosing the optimum initial hyperparameter, requires less computational time, is easy to interpret, and is not affected by the distribution of wireless APs is required. This paper proposes such an algorithm, and the following section provides a detailed description of it.

Table 1. The limitations of fingerprint database clustering algorithms from earlier research

Reference/s	Clustering algorithm	Limitation/s
[10]	DBSCAN and its modifications	Selection of optimum initial parameters (MinPts and Eps); high computational intensity; difficult to interpret
[12, 13]	<i>k</i> -means and its modifications	Selection of the optimum number of clusters and cluster centroids
[15, 17, 18]	APC algorithm and its modifications	High computational intensity; difficult to interpret; and not suitable for all types of fingerprint databases.
[14, 19]	FMC algorithm and its modifications	Selection of the optimum number of clusters and cluster centroids
[11]	SAP algorithm with SAP distance label	Sensitive to the distribution of APs in the environment
[8]	Maximum RSS based algorithm	Poor localization accuracy
Present work	Two nearest APs as similarity measure with an AP label-based vector as cluster centroid	-

3- Proposed Fingerprint Database Clustering Algorithm

The proposed fingerprint clustering algorithm in this paper clusters fingerprints and their corresponding RLs using the two closest APs to each RL as fingerprint similarity measure. The two closest APs to any given RL correspond to the first two APs with the highest RSS values in the RL's fingerprint vector. The AP with the highest RSS value is considered as the AP closest that RL. Also, the AP with the second highest RSS value is the second closest AP to that RL. Using the two closest APs to each RL, fingerprint clusters are generated. The total number of clusters generated is a function of the total number of APs deployed and is calculated using Equation 1.

$$N_{cluster} = N \times (N - 1) \quad (1)$$

Each cluster is labelled using an AP label-based vector using the format shown in Equation 2, and this AP label-based vector serves as the cluster centroid used in determining to which cluster a given fingerprint belongs during the online phase of the localization process.

$$CL_i(x, y) = [x_i, y_i] \quad \text{for } 1 \leq i \leq N_{cluster} \quad (2)$$

where " x_i " and " y_i " denote the labels of the AP with the highest and second highest RSS values, respectively. In the case where the two closest APs to the RL have the same RSS values, " x_i " takes the number of the AP with the smallest label, while " y_i " takes the number of the AP with the highest label. Given a fingerprint database generated using 3 APs labelled #1, #2, and #3, based on Equation 1, there will be a total of 6 clusters. Table 2 shows the AP label-based vector (cluster centroid) of each of the six clusters and the description of the type of fingerprint in each cluster.

Table 2. Cluster centroid label and description of each cluster's fingerprint type

Cluster centroid label	Cluster fingerprint description
$CL_1(1,2) = [1,2]$	Cluster contains fingerprints, with AP #1 having the highest RSS value and AP #2 having the second highest RSS value.
$CL_2(1,3) = [1,3]$	Cluster contains fingerprints, with AP #1 having the highest RSS value and AP #3 having the second highest RSS value.
$CL_3(2,1) = [2,1]$	Cluster contains fingerprints, with AP #2 having the highest RSS value and AP #1 having the second highest RSS value.
$CL_4(2,3) = [2,3]$	Cluster contains fingerprints, with AP #2 having the highest RSS value and AP #3 having the second highest RSS value.
$CL_5(3,1) = [3,1]$	Cluster contains fingerprints, with AP #3 having the highest RSS value and AP #1 having the second highest RSS value.
$CL_6(3,2) = [3,2]$	Cluster contains fingerprints, with AP #3 having the highest RSS value and AP #2 having the second highest RSS value.

Given an arbitrary fingerprint database generated using a total of N APs, a summary of the clustering process of the proposed algorithm is shown below:

Step 1: Using Equation 1, determine the total number of clusters ($N_{cluster}$) to be formed.

Step 2: For each cluster, $1 \leq i \leq N_{cluster}$, create the AP label-based vector centroids using the format in Equation 2.

Step 3: For each RL, identify the label of the APs with the highest and second-highest RSS values.

Step 4: Assign the fingerprint of the RL to the cluster based on the label determined in Step 3.

Step 5: Repeat Steps 3 and 4 for all RLs in the database.

The proposed clustering algorithm requires only three parameters to form clusters and assign fingerprints to each cluster based on the preceding steps, and they are as follows: (1) the fingerprint database; (2) the total number of APs; and (3) the two closest APs to each fingerprint. These three parameters are easily accessible and do not necessitate careful selection.

After the fingerprint database has been clustered using the steps presented previously, it is necessary to know how to determine the cluster in which the fingerprint of an RL is located. This is required for the online phase of the localization process. The magnitude of the element-wise vector subtraction of all the AP label-based vector centroids and the AP label vector generated from the fingerprint is used to determine the cluster in which a fingerprint is located. The cluster whose AP label-based vector centroid results in a zero magnitude is thought to contain that fingerprint. The following is a summary of the procedure for determining in which cluster a fingerprint is located.

Step 1: Given a fingerprint, determine the AP label vector using Equation 3.

$$CL_{fg} = [x_{fg}, y_{fg}] \quad (3)$$

where x_{fg} and y_{fg} are the labels of the APs with the highest and second-highest RSS values, respectively.

Step 2: Find the magnitude of the element-wise vector subtraction of the vector in Equation 3 and all AP label-based vector centroids using Equation 4.

$$\beta_i = \text{sqrt}((x_i - x_{fg})^2 + (y_i - y_{fg})^2) \quad \text{for } 1 \leq i \leq N_{cluster} \quad (4)$$

Step 3: The cluster with $\beta_i = 0$ in Equation 4 is considered to contain the fingerprint.

After identifying the cluster in which the fingerprint is located, the next step is to scan the cluster using the k -NN algorithm to find the RL whose fingerprint matches the real-time acquired fingerprint. The RL obtained is considered to be the IU's estimated instantaneous location from which the fingerprint was obtained. Figure 1 shows an overview of the entire localization process using the proposed clustering algorithm.

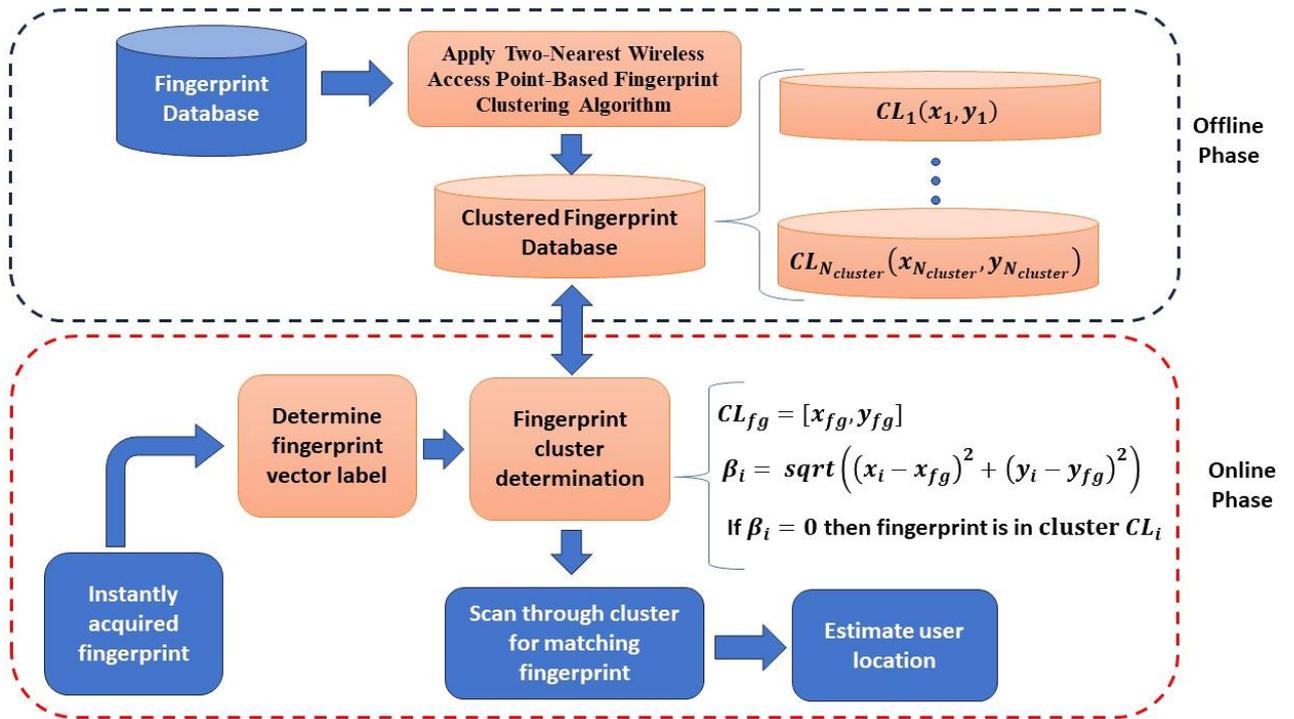


Figure 1. Overview of the localization process with the proposed clustering algorithm

In the next section, the performance of the proposed clustering algorithm is determined and compared with that of the commonly used clustering algorithm with k -NN as the localization matching algorithm.

4- Simulation Results and Discussion

The performance of the proposed clustering algorithm in Section 3 is determined and compared in this section of the paper to other commonly used clustering algorithms. First, the simulation setup and parameters are presented, followed by a performance comparison and discussion.

4-1-Simulation Setup and Parameters

Using two experimentally generated RSS-based fingerprint databases that can be found in Sadowski et al. [23] and Alhmiedat [24], the proposed clustering algorithm's performance is determined. The fingerprint database presented in Alhmiedat [24], which will be referred to as "DB-1", was created with ZigBee as the wireless technology and contains three sub-databases, each created using different RL configurations. The fingerprint database presented in Sadowski et al. [23], which will be referred to as "DB-2," was created with Wi-Fi as the wireless technology and also includes three sub-databases, each created with different RL configurations. Table 3 provides a summary of the characteristics of each of the fingerprint databases considered in this paper.

Table 3. Fingerprint database characteristics

Fingerprint database	Characteristics			
	Wireless technology	No. of APs	Coverage area	Number of RLS
DB-1 [24]	Zigbee	4	Sub-database 1: 21.20 m × 7.60 m	68
			Sub-database 2: 21.20 m × 7.60 m	126
			Sub-database 3: 21.20 m × 7.60 m	194
DB-2 [23]	Wi-Fi	3	Sub-database 1: 5.8 m × 5.3 m	16
			Sub-database 2: 10.8 × 7.3 m	40
			Sub-database 3: 6.0 m × 5.5 m	49

The performance of the proposed clustering algorithm is determined and compared for each fingerprint database and its sub-databases with the widely used fingerprint clustering algorithms, FCM and k -means. Furthermore, the proposed clustering algorithm will also be compared with the lightweight maximum RSS clustering (LMRC) algorithm developed and presented in Quezada-Gaibor et al. [8]. For the localization matching algorithm, the k -NN algorithm is considered with $k = 3$, which is considered to be the optimum " k " value [22]. Based on previous research, the optimal number of clusters for k -means and FCM algorithms was found to be 3 [13]. The performance comparison of all clustering algorithms is carried out using a computer with the following characteristics: an Intel (R) Core (TM) i5-2400 CPU at 3.10 GHz, 12 GB of RAM, the Windows 10 operating system, and MATLAB R2020a.

4-2-Localization Performance Comparison

This subsection compares the performance of the proposed algorithm to that of the FCM, k -means, and LMRC algorithms using the two databases (DB-1 and DB-2), each with the characteristics listed in Table 3. The performance comparison in terms of localization time is first presented, followed by the performance comparison in terms of localization accuracy using position root mean square (RMSE) as a performance metric.

4-2-1- Localization Time Comparison

Given the instantly acquired fingerprint measurement, the time it takes for the system to scan through the fingerprint database to find the RL (IU location) with the matching fingerprint is known as the localization time. The system localization time is very important as it determines the real-time applicability of that system. For every RL in the two fingerprint databases, the localization time is determined. Table 4 shows the average localization time of all RLS in each fingerprint database sub-database using FCM, k -means, LMRC, and the proposed clustering algorithm. Figure 2 shows a graphical comparison of the average localization time.

Table 4. Average localization time Comparison

Algorithm	Average localization time (10^{-4} sec)					
	DB-1			DB-2		
	Sub-database 1	Sub-database 2	Sub-database 3	Sub-database 1	Sub-database 2	Sub-database 3
FCM	2.27	2.27	2.31	2.27	2.27	2.25
k -means	2.06	2.10	2.14	2.02	2.05	2.04
LMRC	2.62	2.66	2.74	2.60	2.69	2.65
Proposed	1.13	1.14	1.17	1.10	1.12	1.12

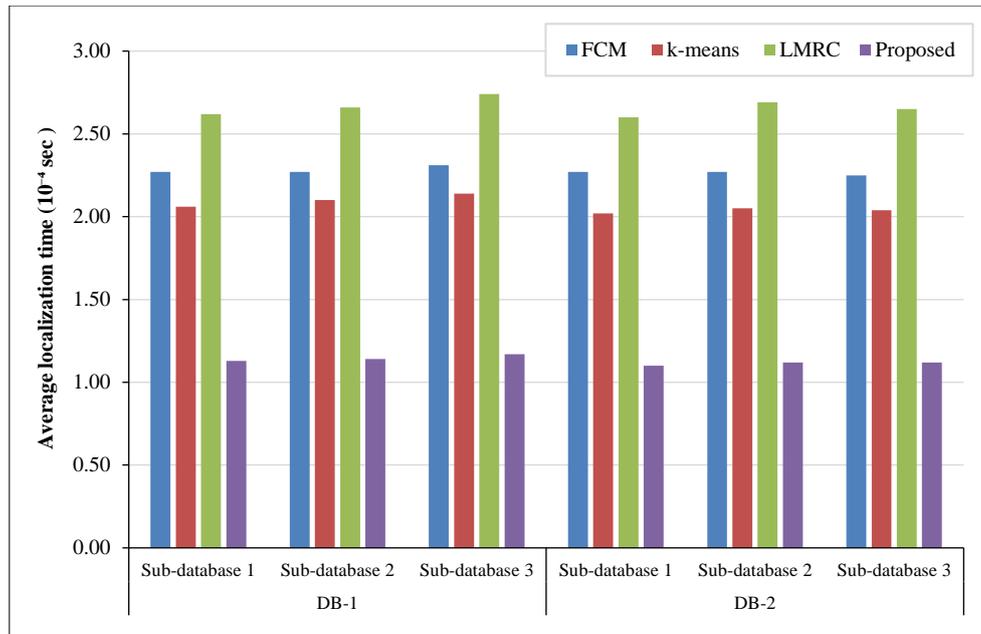


Figure 2. Localization time comparison for all Subcluster and databases

The average localization time for all four clustering algorithms is approximately the same when comparing the results from the two databases (DB-1 and DB-2). For instance, for the FCM algorithm, the average localization time for DB-1 considering all three sub-databases is 2.28×10^{-4} sec, and the average localization time for DB-2 considering all three sub-databases is 2.27×10^{-4} sec. The localization time difference between the two databases for the FCM algorithm is 0.02×10^{-4} sec, corresponding to a 0.8% difference. The average localization time percentage differences between DB-1 and DB-2 considering all three sub-databases for the *k*-means, LMRC, and proposed algorithm are 2.8, 0.7, and 3.4%, respectively. This means that, despite differences in fingerprint database characteristics, the average localization time for each algorithm in the two databases is approximately the same. As such, any conclusions in terms of localization time derived from either of the databases are assumed to be the same for the other database.

Looking at the localization time comparison for DB-1 for all four algorithms, it can be seen that the proposed clustering algorithm has the fastest localization time for all three sub-databases, averaging to about 1.15×10^{-4} sec. The next fast algorithm is the *k*-means algorithm, with an average localization time across the three sub-databases of about 2.1×10^{-4} sec. This is followed by the FCM algorithm with an average localization time of 2.28×10^{-4} sec, and the slowest is the LMRC algorithm with an average localization time of 2.67×10^{-4} sec across all three sub-databases. On average, considering all two databases and the three sub-databases of each database, the proposed algorithm is about 50%, 45%, and 57% faster in localising an IU given the fingerprint than the FCM, *k*-means, and LMRC algorithms, respectively.

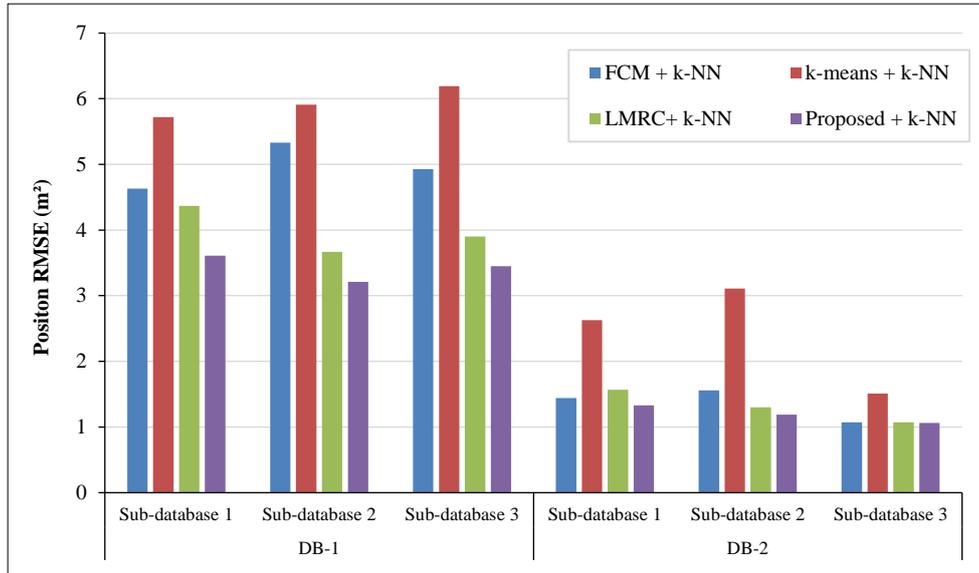
In summary, considering all two databases and their sub-databases, the proposed clustering algorithm is at least 45% faster in terms of localization time. One of the objectives of any real-time localization system is to be able to localise an IU in near real-time, and from the results in Table 4, the proposed algorithm is the best choice to achieve that. This is because the clustered fingerprint database based on the proposed clustering algorithm resulted in a very fast localization time using the *k*-NN algorithm. However, fast localization time does not necessarily translate to accurate localization. In the next subsection, the localization accuracy of the proposed algorithm is determined and compared with the LMRC, FCM, and *k*-means algorithms.

4-2-2- Localization Accuracy Comparison

In the previous subsection, it was established that the use of the proposed clustering algorithm to cluster fingerprint databases results in a very fast localization time. However, it is important to know how accurate the IU locations are obtained using the clustered fingerprint database of each clustering algorithm when paired with the *k*-NN algorithm. Localization accuracy is also a measure of the accuracy and efficiency with which a clustering algorithm clusters a fingerprint database. The position RMSE at which all RLs are obtained given the input fingerprints is determined and presented in Table 5 using the clustered fingerprint database of each clustering algorithm. Figure 3 shows a graphical comparison of the position RMSE of the four clustering algorithms.

Table 5. Position RMSE Comparison

Algorithm	Position RMSE (m ²)					
	DB-1			DB-2		
	Sub-database 1	Sub-database 2	Sub-database 3	Sub-database 1	Sub-database 2	Sub-database 3
FCM + <i>k</i> -NN	4.63	5.33	4.93	1.44	1.56	1.07
<i>k</i> -means + <i>k</i> -NN	5.72	5.91	6.19	2.63	3.11	1.51
LMRC+ <i>k</i> -NN	4.37	3.67	3.90	1.57	1.30	1.07
Proposed + <i>k</i> -NN	3.61	3.21	3.45	1.33	1.19	1.06

**Figure 3. Position RMSE comparison for all Subcluster and databases**

Even though the localization time achieved by each clustering algorithm for the two fingerprint databases is the same, the accuracy at which IU locations are obtained for the two databases is different. The position RMSEs obtained by all four algorithms when paired with the *k*-NN algorithm are lower for DB-2 when considering all three sub-databases. The average position RMSE obtained by the FCM + *k*-NN algorithm when applied to all three sub-databases of DB-1 is 5.0 m², while that of DB-2 considering all three sub-databases is 1.36 m². This corresponds to an average percentage difference of about 73%. The average percentage differences in position RMSE achieved by *k*-mean + *k*-NN, LMRC + *k*-NN, and proposed + *k*-NN algorithms are 60%, 67%, and 65%, respectively. The significant differences between the two databases are the total number of RIs and APs per sub-cluster. The number of APs and RIs in all three sub-clusters in DB-1 is, on average, higher than that in DB-2. This resulted in higher position RMSE values for all clustering algorithms in DB-1.

Looking at the position RMSEs of all the algorithms considering all three sub-databases in DB-1, the proposed + *k*-NN algorithm has, on average, the least position RMSE of about 3.42 m². The next least position RMSE, which is 3.98 m², is obtained by the LMRC + *k*-NN algorithm, followed by 4.96 m² obtained by the FCM + *k*-NN algorithm. The *k*-means + *k*-NN algorithm has the highest position RMSE of about 6.19 m². This means that the proposed + *k*-NN algorithm has the best localization accuracy in DB-1, which is 31, 42, and 14% higher than the FCM + *k*-NN, *k*-means + *k*-NN, and LMRC + *k*-NN algorithms, respectively. On average, the proposed + *k*-NN algorithm has a localization accuracy that is about 29% higher than the other three algorithms considered when applied to the DB-1.

Extending the analysis to DB-2, the proposed + *k*-NN algorithm also has the least average position RMSE of about 1.19 m² considering all three sub-databases. This is followed by the LMRC + *k*-NN algorithm with an average position RMSE of 1.13 m², the FCM + *k*-NN algorithm with 1.36 m², and the *k*-means + *k*-NN algorithm with 1.51 m². The proposed + *k*-NN algorithm achieved localization accuracy improvements of about 13%, 51%, and 9% over the FCM + *k*-NN, *k*-means + *k*-NN, and LMRC + *k*-NN algorithms, respectively. On average, the proposed + *k*-NN algorithm has an average localization accuracy improvement of about 24% when compared to the other three clustering algorithms in DB-2.

The discussion so far has shown that, when applied to both DB-1 and DB-2, the proposed + *k*-NN algorithm outperformed the other three algorithms by at least 24% in terms of localization accuracy and 45% in terms of localization time. This shows the proposed clustering algorithm's superiority in terms of clustering efficiency and

accuracy over the k -means, FCM, and LMRC algorithms. The proposed clustering algorithm does not require the optimal selection of initial hyperparameters as k -means and FCM do, and it is not sensitive to AP distribution as the LMRC algorithm is. Overall, these findings suggest that the proposed clustering algorithm could be a useful tool for improving indoor localization accuracy in a variety of applications, such as indoor navigation, asset tracking, and location-based services.

5- Conclusion

One of the limiting factors in the performance of a fingerprint-based localization system is the density of the fingerprint database. The higher the density, the higher the localization accuracy; however, the longer the localization time. Clustering algorithms have been used to overcome this trade-off. However, conventionally used clustering algorithms suffer from several limitations, some of which are the need to select optimum initial hyperparameters such as cluster number and cluster centroid, high computational intensity, and being very difficult to interpret.

This paper presents an algorithm for clustering fingerprint databases aimed at improving localization time and accuracy while at the same time not requiring the selection of an optimum initial parameter and being computationally light. The algorithm clusters fingerprints using the two closest APs to the RL at which the fingerprint is obtained and the AP label-based vector as the cluster centroid. The performance of the proposed algorithm in terms of localization time and accuracy was compared to the k -means, FCM, and LMRC algorithms, with the k -NN algorithm as the localization algorithm. The result obtained shows that the proposed algorithm is at least 45% faster in localization time and 25% higher in localization accuracy compared to the other three algorithms. These findings suggest that the proposed algorithm is efficient and accurate at clustering fingerprint databases, and with its lower localization time, it has real-time application capability. However, the presence of outlier fingerprints has a significant effect on the clustering accuracy of any clustering algorithm. In future research, the performance of the proposed clustering algorithm will be determined using fingerprint databases with outliers, and possible solutions to overcome this performance limiting factor will be proposed.

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, F.M.; project administration, 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 project 2023 run at the Faculty of Informatics and Management, University of Hradec Kralove, Czech Republic.

6-4- Acknowledgements

The authors would like to thank the management of the University of Hradec Králové (UHK) in the Czech Republic, as well as the Faculty of Informatics and Management (FIM), for providing the resources and support for this study. Ing. Kruncik's technical assistance is also gratefully acknowledged.

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., 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.
- [2] Yang, T., Cabani, A., & Chafouk, H. (2021). A survey of recent indoor localization scenarios and methodologies. *Sensors*, 21(23), 8086. doi:10.3390/s21238086.
- [3] Shang, S., & Wang, L. (2022). Overview of WiFi fingerprinting-based indoor positioning. *IET Communications*, 16(7), 725–733. doi:10.1049/cmu2.12386.
- [4] Kriz, P., Maly, F., & Kozel, T. (2016). Improving Indoor Localization Using Bluetooth Low Energy Beacons. *Mobile Information Systems*, 2016, 1–11. doi:10.1155/2016/2083094.
- [5] Fronckova, K., & Prazak, P. (2020). Possibilities of using Kalman filters in indoor localization. *Mathematics*, 8(9), 1564. doi:10.3390/math8091564.
- [6] Alhomayani, F., & Mahoor, M. H. (2020). Deep learning methods for fingerprint-based indoor positioning: a review. *Journal of Location Based Services*, 14(3), 129–200. doi:10.1080/17489725.2020.1817582.
- [7] 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.
- [8] 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)*. doi:10.1109/ipin51156.2021.9662612.
- [9] Gomes, E. L., Fonseca, M., Lazzaretti, A. E., Munaretto, A., & Guerber, C. (2022). Clustering and Hierarchical Classification for High-Precision RFID Indoor Location Systems. *IEEE Sensors Journal*, 22(6), 5141–5149. doi:10.1109/JSEN.2021.3103043.
- [10] Marakkalage, S. H., Lau, B. P. L., Zhou, Y., Liu, R., Yuen, C., Yow, W. Q., & Chong, K. H. (2021). WiFi Fingerprint Clustering for Urban Mobility Analysis. *IEEE Access*, 9, 69527–69538. doi:10.1109/ACCESS.2021.3077583.
- [11] 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.
- [12] 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)*. doi:10.1109/vtc2021-spring51267.2021.9448961.
- [13] Zhao, H. (2022). Design and Implementation of an Improved K-Means Clustering Algorithm. *Mobile Information Systems*, 2022, 1–10. doi:10.1155/2022/6041484.
- [14] Sun, Y., Xu, Y., Ma, L., & Deng, Z. (2009). KNN-FCM hybrid algorithm for indoor location in WLAN. *2009 2nd International Conference on Power Electronics and Intelligent Transportation System (PEITS)*. doi:10.1109/PEITS.2009.5406793.
- [15] Subedi, S., Gang, H. S., Ko, N. Y., Hwang, S. S., & Pyun, J. Y. (2019). Improving indoor fingerprinting positioning with affinity propagation clustering and weighted centroid fingerprint. *IEEE Access*, 7, 31738–31750. doi:10.1109/ACCESS.2019.2902564.
- [16] Wu, Y., Jia, Z., Dai, Y., & Wang, W. (2022). Research on WiFi Location Fingerprint Positioning Algorithm Based on DPC-FCM Clustering. *Wireless Communications and Mobile Computing*, 2022, 1–7. doi:10.1155/2022/8347084.
- [17] Nurhidayat, I., Pimpunchat, B., Noeiaghdam, S., & Fernández-Gámiz, U. (2022). Comparisons of SVM kernels for insurance data clustering. *Emerging Science Journal*, 6(4), 866–880. doi:10.28991/ESJ-2022-06-04-014.
- [18] 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.
- [19] Hung, S. L., Kao, C. Y., & Huang, J. W. (2022). Constrained K-means and Genetic Algorithm-based Approaches for Optimal Placement of Wireless Structural Health Monitoring Sensors. *Civil Engineering Journal*, 8(12), 2675–2692. doi:10.28991/CEJ-2022-08-12-01.
- [20] Nagaraj, P., Birunda, S. S., Venkatesh, R., Muneeswaran, V., Narayanan, S. K., Shree, U. D., & Sunethra, B. (2022). Automatic and Adaptive Segmentation of Customer in R framework using K-means Clustering Technique. *2022 International Conference on Computer Communication and Informatics (ICCCI)*. doi:10.1109/iccci54379.2022.9741067.
- [21] 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. *2020 12th International Congress on Ultra-Modern Telecommunications and Control Systems and Workshops (ICUMT)*. doi:10.1109/icumt51630.2020.9222458.
- [22] Caso, G., de Nardis, L., & di Benedetto, M. G. (2015). A Mixed approach to similarity metric selection in affinity propagation-based WiFi fingerprinting indoor positioning. *Sensors (Switzerland)*, 15(11), 27692–27720. doi:10.3390/s151127692.
- [23] 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.
- [24] Alhmiedat, T. (2023). Fingerprint-Based Localization Approach for WSN Using Machine Learning Models. *Applied Sciences (Switzerland)*, 13(5), 3037. doi:10.3390/app13053037.