DTU Library

A Weighted K-AP Query Method for RSSI based Indoor Positioning

Huo, Huan; Liu, Xiufeng; Li, Jifeng; Yang, Huhu; Peng, Dunlu ; Chen, Qingkui

A Weighted K-AP Query Method for RSSI based Indoor Positioning

Huan Huo[1], Xiufeng Liu[2], Jifeng Li[3], and Huhu Yang[1]

[1] School of Optical-Electrical and Computer Engineering
University of Shanghai for Science and technology, Shanghai 200093, China
[2] Technical University of Denmark
[3] University of Oulu, FI-90014, Finland

Abstract. The paper studies the establishment of offline fingerprint library based on RSSI (Received Signal Strength Indication), and proposes WF-SKL algorithm by introducing the correlation between RSSIs. The correlations can be transformed as AP fingerprint sequence to build the offline fingerprint library. To eliminate the positioning error caused by instable RSSI value, WF-SKL can filter the noise AP via online AP selection, meanwhile it also reduces the computation load. WF-SKL utilizes LCS algorithm to find out the measurement between the nearest neighbors, and it proposes K-AP (P,Q) nearest neighbor queries between two sets based on Map-Reduce framework. The algorithm can find out K nearest positions and weighted them for re-positioning to accelerate the matching speed between online data and offline data, and also improve the efficiency of positioning. According to a large scale positioning experiments, WF-SKL algorithm proves its high accuracy and positioning speed comparing with KNN indoor positioning.

1 Introduction

As the rapid growth of wireless networking and the development of portable devices, a new upsurge of LBS (Location Based Service) applications are in making [1].

Nowadays the LBS has several ways to deal with the Indoor Positioning Service(IPS), such as Bluetooth[2], IR (Infrared Radiation)[3], RF (Radiation Frequency)[3], ultrasound [4] and UWB (Ultra-Wide Bandwidth)[5]. Several of them rely on electromagnetic wave, nevertheless, the electromagnetic wave spreading can be influenced by multipath effect. As the reflection and diffraction caused by the materials and the shapes of interior architecture affect the wave spreading, the signal error would be very high[6]. The Ultrasound and UWB could reach the accuracy within 1cm [7], however, the high demand of hardware needs a lot of equipments to support, which brings extremely high cost, hence they are not suitable for commercial use. WIFI infrastructures are widely deployed in most shopping centers and supermarkets, and also have relative high accuracy and low cost, therefore WIFI is a popular solution for both academic research and industrial application.

In reality, IPS could be based on either Bluetooth or WIFI signal strength. Currently, WIFI-based IPS could utilize trilateration-centroid algorithm[8], geographic information probability [1] or RSSI-based algorithm [9]. Furthermore, the researchers proposed two matching algorithms based on uncertainty and

probability being built on RSSI-based algorithm. The uncertainty-based matching algorithm adopts KNN algorithm deriving from Kalman filtering, and the probability-based matching algorithm adopts Bayes probability [10].

Due to the unstable RSSI, WIFI-based indoor positioning still has a great gap of accuracy with costly UWB and ultrasonic wave positioning. The complex indoor architecture caused multipath effect, reflection and diffraction, hence the key point is to improve the stability of signal. For this issue, a general solution is to build a fingerprint library.

According to [5], the researcher proposes a method of building the fingerprint library depending on RSSI values as the clustering pattern (AP signal strength). Although the fingerprint library increases the efficiency of the algorithm via clustering, the data error derived from instable RSSI value, which is called jitter, generates sampling error and reduces the accuracy of position estimation. Also, the calculation of RSSI value is too complex, and it consumes huge amount of store space.

The researchers mentioned uncertainty in [11]. The signal strength sampled at one place will change and fluctuate in a small range with the time pass by. An experiment proves that the sequences of a pair of AP would be stable if the RSSI value exceed 10dBm between them[12].

In this paper, we propose an improved indoor WIFI positioning algorithm based on fingerprint library and weighted KNN. Comparing with traditional method which only utilize the uncertain RSSI value, the new method collects fingerprints from APs signal and introduces the *AP* correlation into the algorithm. Moreover, to form a stable sequence-pair, the new method sorts the signals by the strengths from strong to weak, and the sequence-pair solve the issue caused by RSSI fluctuation. Considering the features of APs sequence, we introduce the concept of AP-based reference point(RP) to improve the efficiency of the algorithm. Finally we propose the weighted k-AP(P,Q) query algorithm. The experiment proves that the positioning accuracy is improved after introducing the correlation of the APs RSSI into the algorithm, meanwhile the positioning speed is faster with the fingerprint library clustering.

2   Preliminaries

Before introducing the IPN algorithm, we outlines our framework by introducing the following concepts and definitions.

Definition 1. Wireless Access Point (AP)
An AP is a networking hardware device that allows a Wi-Fi compliant device to connect to a wired network. In this paper, AP refers to the physical location where Wi-Fi access to a WLAN is available.

Definition 2. Reference Point (RP)
In the stage of building offline fingerprint library, we need to collect the signal strength value from a series of APs as the control data. These points are considered reference points.

Definition 3. Longest Common Subsequence (LCS)
LCS is used to measure the length of non-continuous longest common subsequence between two ordered sequences, which represents the similarity between two sequences. In this paper, LCS refers to the number of overlapping APs between online AP sequences and offline RP fingerprint of AP sequences. The more overlapping APs, the higher degree of matching between two points.

Definition 4. Received Signal Strength Indication (RSSI)

Received Signal Strength Indication is the criteria to evaluate the quality of connection and signal strength in sending process. It estimates the distance between the sender and the receiver depending on the strength of the received signal, furthermore to position the location based on relevant data.

2.1 Online AP Selection

Due to the difference and variations between the environments of online and offline data sampling, we need to filter out the APs which are affected by environment significantly, to improve the accuracy of matching. Assuming only a few APs will be influenced by the environment at any certain time point, in other words, part of data is expired in the online testing, hence we need to introduce AP selection mechanism. For example, in a big scale indoor environment, some APs are operated by ISP as public WIFI APs, and some others are temporary individual APs. We can define the APs in this area as a set $R = R_1, R_2, \ldots, R_m, |R| = M$. Via the AP selection algorithm such as MaxMean[16], InfoGain, and Deccorelated space, we can select a new AP set for positioning and delete the invalid APs. The principle of AP selection is to find a subset A which cannot influence the outcome of positioning, meanwhile adding new APs into subset A would not improve the outcome of positioning. We define the subset $A = A_1, A_2, \ldots, A_L, |A| = L, L < M$. This paper adopts RANSAC [5] as the AP selection algorithm. It filters out noise AP being influenced by the environment via probability distribution histogram testing. It improves the accuracy and reduce the calculation based on large scale offline data of APs.

2.2 RSSI for Indoor Positioning

Briefly, the fingerprint library indoor positioning algorithm is a type of RSSI-based KNN algorithm. The principle of the algorithm is to utilize the value of signal strength as the attribute of RP after the RW device receives the wireless signal strength being sent from the AP, then we can use KNN to find out number K of reference points, and adopt the locations of reference points to estimate the query point position.

RSSI based indoor positioning has two steps. The first one is to build the offline fingerprint library, and the second step is to apply the KNN matching algorithm on online data. Building offline fingerprint library is the stage of APs signal sampling via selecting huge amount of RPs, as shown in Fig.1.

3 A Weighted K-AP(P,Q) Query Method

3.1 Fingerprint Library Construction

Assume in an indoor environment, the numbers of APs and RPs are M and N separately, noted as $m = 1, 2, 3, , M$ and $n = 1, 2, 3, , N$. For each RP, we sample Q times and define the ith RP sampling from the APs as $S_n^q = (S_{n1}^q, S_{n2}^q, , S_{nM}^q)$. So the RSSI matrix for the RPs after Q times sampling is defined as S_n.

$$S_n = \begin{pmatrix} S_n^1 \\ S_n^2 \\ \vdots \\ S_n^Q \end{pmatrix} = \begin{pmatrix} S_{n1}^1 & S_{n2}^1 & \cdots & S_{nM}^1 \\ S_{n2}^2 & S_{n2}^2 & \cdots & S_{nM}^2 \\ \vdots & \vdots & \ddots & \vdots \\ S_{n1}^Q & S_{n2}^Q & \cdots & S_{nM}^Q \end{pmatrix} \tag{1}$$
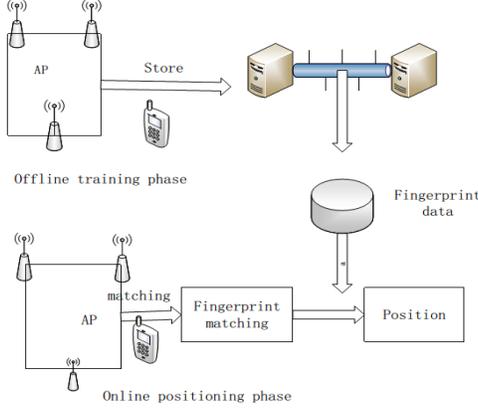
Fig. 1. RSSI based Indoor Positioning

In the original RSSI data, the accuracy for weak signals is very low due to the device error, or operative error. For different RPs, some are always receiving weak signal from certain AP, and their RSSI accuracy can not be guaranteed by the device if the RSSI is below the lower bound. Therefore, RSSI matrix can not indicate the signal strength among different APs. In order to reduce the data storage and unnecessary computing, we introduce $NoiseAP$ for offline processing to improve the analysis accuracy. As for the fingerprint of each sampling, we set the $NoiseAP$ with $-\infty$ so as to avoid the different lengths for each data set.

Definition 5. Noise AP
An AP_i is a $NoiseAP$ according to the weakness of the received signal strength, if $S_{ni} \leq \theta$. θ is the lower bound of the signal strength received by the device.

Then we introduce the correlation among different RSSI. As the signal strengths received from different AP are different, the closer the AP, the stronger the signal. So it would be inaccurate if the estimation solely relies on the individual AP signal strength. In order to better reflect the relative position of the RPs, we rank APs according to the signal strength. Thus for each data set S_{nm}^q in the RSSI matrix S_n, we generate a ranked $\overline{S}_{nm}^q$ with corresponding AP ID. The transformed matrix of AP is depicted as below:

$$\overline{S}_n = \begin{pmatrix} \overline{S}_n^1 \\ \overline{S}_n^2 \\ \vdots \\ \overline{S}_n^Q \end{pmatrix} = \begin{pmatrix} AP_{n1}^1 & AP_{n2}^1 & \cdots & AP_{nM}^1 \\ AP_{n2}^2 & AP_{n2}^2 & \cdots & AP_{nM}^2 \\ \vdots & \vdots & \ddots & \vdots \\ AP_{n1}^Q & AP_{n2}^Q & \cdots & AP_{nM}^Q \end{pmatrix} \tag{2}$$

Where $AP_{ni}^q = m$ represents in the $q's$ sampling, for the nth RP, the ith ranked AP is marked with ID m.

Our goal is not for the Qth sampling data for single RP, but to generate a fingerprint according to the common feature of the RP on RSSI from Q times of sampling. Therefore, our approach is to calculate the the ID with highest frequency on each column, and set the IDas the representative of the column, which means that through Q times sampling, the AP of the fingerprint on this column is ID. For example, after Q times sampling and transformation, the AP fingerprint of the $nthRP$ is $\bar{S}_n = (M, 20, 18, \ldots, 1)$, assuming that the signal strength from AP M is the strongest one, then the AP with ID 20 follows, and so on.

However, there are some special circumstances. For instance, assuming the AP fingerprint for certain RP is $\bar{S}_i = (38, 20, 20, 25, 9, 10, \ldots, 3)$, there are two identical IDs. This is because the interference during the signal transmission, or severe error from measurement. The phenomenon will be eliminated along with the increment of sampling times. The smaller the Q, the higher possibility the repeated AP appears. In theory, the phenomenon can be fully avoided given unlimited Q. But in reality, it is impossible to sample infinite times for large amount of AP.

So we propose a solution for calculating the representative AP. Firstly, we locate the columns with repeated AP, compare the repeated frequency of the APs on each column, and set the AP as the representative AP if its frequency is greater than others. Then we set the AP with the second largest frequency as the representative AP for the next column, and iterate the same process until there is no repeated AP on each column.

For instance, the AP with ID 20 apears m times on the second column, while the the AP with ID 20 apears n times on the third column. If $m > n$, we keep the AP with ID 20 as the representative for the second column, and substitute the second frequent AP with ID 18 as the representative for the third column. Finally, we can get the AP fingerprint of the RP as $\bar{S}_i = (38, 20, 18, 25, 9, 10, \ldots, 3)$.

Based on the above approach, the RSSI matrix for all the RPs can be transformed to the AP fingerprint library for all the RPs. We name the offline fingerprint library as S'.

$$S' = \begin{pmatrix} \overline{S}_1 \\ \overline{S}_2 \\ \vdots \\ \overline{S}_N \end{pmatrix} = \begin{pmatrix} AP_{11} & AP_{12} & \cdots & AP_{AM} \\ AP_{21} & AP_{22} & \cdots & AP_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ AP_{N1} & AP_{N2} & \cdots & AP_{NM} \end{pmatrix} \tag{3}$$

At last, we analyse the AP fingerprint matrix and perform our clustering methods according to the first column fingerprint, i.e. the strongest signal AP for each RP. For example, if RP_i and RP_j share the same strongest AP from M, we cluster RP_i and RP_j together and marked the APs ID asC_M. Then we get the fingerprint cluster, as shown below.

$$S'' = \begin{pmatrix} C_1 \\ C_2 \\ \vdots \\ C_M \end{pmatrix} \tag{4}$$

The fingerprint cluster can speed up the matching efficiency by fast locating the similar RPs when matching the online data and the offline data, so as to improve the positioning efficiency.

3.2 A Weighted K-AP(P,Q) Query Method based on MapReduce

In traditional K-NN algorithm, we can calculate the top K nearest RPs by computing the Euclidean distance between the query point and the offline points, and then estimate the location by the means of their coordinates. With the proposed AP fingerprint library, we do not need to calculate the Euclidean distance to estimate the similarity between the query point and the offline vectors. Instead, we estimate the similarity by simply counting the number of the overlapped data between the two vectors. The similarity is higher if the two vectors share more common data. Therefore, we calculate the longest common subsequence by the LCS equation below:

$$LCS(s'_m, \bar{S}_n) = \begin{cases} \emptyset & if m = 0 \, or \, n = 0 \\ LCS(s'_{m-1}, \bar{S}_{n-1}) & if ap_m = AP_n \\ max(LCS(s'_m, \bar{S}_{n-1}), LCS(s'_{m-1}, \bar{S}_n)) & if ap_m = AP_n \end{cases} \tag{5}$$

In Equation 5, $LCS(s'_m, \bar{S}_n)$ is the common AP length between the online s'_{m_i} and offline $\bar{S}_n$. s'_{m-1} and $\bar{S}_{n-1}$ are the remained parts after removing the first common AP. $ap_m = AP_n$ represents the overlapped AP in the two sequences. When one of the sequences contains no AP, $LCS(s'_m, \bar{S}_n))$ is set to 0. If both sequences contain at least one AP, we recursively compute the maximum LCS. In the algorithm, similar sub sequences are not mandatory continuous.

For instance, assume the fingerprint of the query RP is $s = (s_1, s_2, ..., s_M)$, the AP fingerprint is $s' = (ap_1, ap_2, , ap_M)$ after ranking, and the fingerprint vector for ith RP is $\bar{S}_i = (AP_{i1}, AP_{i2}, , AP_{iM})$. Then the similarity of $LCS(s'_m, \bar{S}_n)$ is computed as l.

When estimating the location, we propose a K-AP(P,Q) query algorithm for finding the top k nearest AP pair between two data sets P and Q, based on the $K - NN$ query on MapReduce platform. The main idea is to build a R*-tree index for data set P and Q, then perform K-AP(P,Q) query according to the similarity definition on MapReduce. As MapRedcuce parallels the original K-NN algorithm, the K-AP(P,Q) algorithm can efficiently find the best matching out of the offline fingerprint library.

The data partition strategy is to locate the strongest AP of the online data into the cluster C_i, and construct set P for all the AP fingerprints in C_i. While set Q contains only the online data. Then we construct the parallel R*-tree index based on the sampling procedure and Hilbert curve construction [13]. The details of the index construction can be referred to [14].

According to the K-AP(P,Q) algorithm, we get the top K nearest RPs, based on which we can calculate the query position with the Equation 6.

$$\begin{cases} (X, Y) = & \dfrac{1}{K} \sum_{i=1}^{K} w_i(s_i, y_i) \\ \sum_{i=1}^{k} w_i = & 1 \end{cases} \tag{6}$$

In Equation 6, (x_i, y_i) is the real coordinates of the ith RP, and w_i is the weight of the ith RP. We redefine the weight to improve the estimation accuracy as shown in Equation 7.

$$w_i = \frac{l_j}{\sum_{j=1}^{k} l_j} \qquad (7)$$

Equation 7 indicates that the location similarity will increase along with the increment on the overlapping parts between the query point and the reference points. Based on the above analysis, we propose the WF-SKL algorithm on fingerprint library for the indoor positioning, as shown in Algorithm 1.

Algorithm 1 WF-SKL

Input: (1) N RPs, M RP coordinates, Sampling time Q;
    (2) Original RSSI fingerprint set S;
    (3) Online RSSI fingerprint s;
    (4) Noise AP threshold θ, rank k.
Output: Current point position coordinates: (x, y).
1: Generate M APs from AP selection
2: **for** each RP **do**
3:    collect RSSI from each AP
4:    **if** RSSI $< \theta$ **then**
5:       set $-\infty$
6:    **end if**
7:    get S
8: **end for**
9: **for** each RP in S **do**
10:    **for** each time in S_i **do**
11:       rank (S_i^q)
12:       return $\bar{S}_i$
13:       **while** There exist at least one pair of the same AP in the $\bar{S}_i$ **do**
14:          reallocateAP();
15:       **end while**
16:       return S'
17:    **end for**
18: **end for**
19: cluster (S')
20: return S''
21: // K-AP Query
22: K-AP(P, Q)
23: select C_i as P, s as Q
24: return k APs
25: // Positioning Estimation
26: do $\begin{cases} (X, Y) = \frac{1}{K} \sum_{i=1}^{k} w_i (x_i, y_i) \\ \sum_{i=1}^{k} w_i = 1 \end{cases}$.
27: return (x, y)

---

4   Experiment

We did a bunch of experiments to evaluate the performance of the indoor positioning methods based on AP fingerprint library and the weighted $K - AP(P, Q)$

query algorithm. The experiments took place on the sixth floor of the University of Shanghai for Science and Technology. The experiment area was 566 $m^2(35.6m \times 15.9m)$, located with 16 TP-LINK TL-WDR4300 N750 double band wireless routers functioning as AP. The RSSI values were detected through a Sumsang Note 4 smart phone based on Android 5.0 and collected by the application ADAwifi. We randomly selected 50 offline RPs and 20 online testing points. Each RP was sampled 30 times. Fig.2 shows the distribution of all the RPs, testing points and the APs.
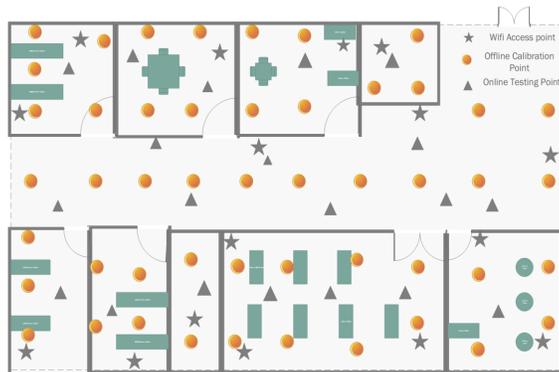

Fig. 2. Distributions of APs, RPs and Testing Points

The PC configuration for data analysis is Intel(R) Core(TM) i3 M370 with 2.4GHz CPU, 500Gbyte hard disk, and Windows8 system. MySQL is the offline storage platform.

4.1 AP Selection Performance

First, in order to verify the influence of the online AP selection, our experiment simulated two groups of testing point positioning in the same environment. In the experiment, we applied WF-SKL algorithm to two groups, one of which directly queried the position while the other performed online AP selection first. Fig. 3 shows the outcome comparison of the positioning error.

In Fig. 3, the positioning error is smaller if the method performs AP selection first. This is because the AP selection filters out the individual AP deviation caused by the environment vibration. Thus the AP selection helps to improve the positioning accuracy.

4.2 Accuracy Comparison

Then we performed the experiments for the offline AP fingerprint library construction. In the same simulation environment, we applied AP selection first and then estimated the position by the weighted $K - AP(P, Q)$ algorithm, and compared our method with the traditional KNN based fingerprint construction
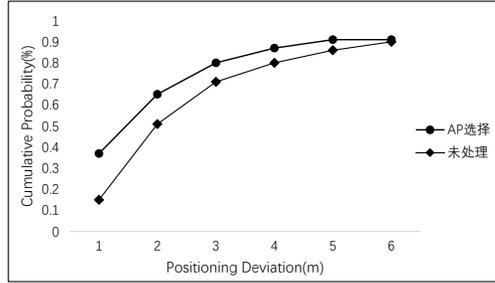
Fig. 3. AP Selection Performance

and the KNN based on clusters fingerprint construction [5]. The accuracy comparison outcome is shown in Fig. 4 and the performance comparison outcome is shown in Fig. 5:
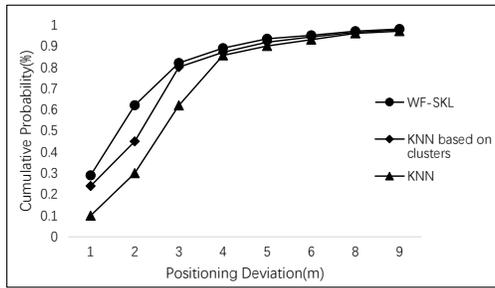

Fig. 4. Accuracy Comparison

As seen from Fig.4, the accuracy of WF-SKL algorithm is much better than the traditional KNN algorithm. The reason why our fingerprint computation method is more accurate than the method is because ours reduces the errors generated by rough estimates rather than simply computing the average. We also found that our algorithm is more accurate than the KNN based on clusters algorithm. This is because KNN based on clusters algorithm selects the strongest RSSI cluster, but its data is based on single RSSI value, thus can not prevent the errors caused by the unstable RSSI. Our method is a relatively stable AP pairs positioning estimation, which can avoid the errors caused by using individual RSSI values.

Fig.5 indicates the positioning speed of different algorithms. In the real environment, the scale of fingerprint library increases with the declining distance between the testing points, which means the positioning accuracy based on fingerprint library would be higher. Fig. 5 shows the KNN algorithm has the lowest processing speed. It does not cluster the data in the offline stage, hence the calculation in the stage of online matching slows down the speed. WF-SKL algorithm has a faster speed because it adopts AP pair as matching data, which is simpler
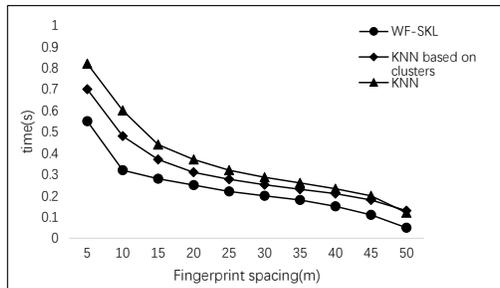
Fig. 5. Performance Comparison

than RSSI being used by KNN based on clusters as matching data. Not like KNN based on clusters algorithm calculating RSSI as the matching data, WF-SKL algorithm calculates the common sequence via LCS to process the matching query, therefore the matching speed is faster.

4.3   Weight Evaluation

We tested the performance of weighting in this experiment. The testing dataset adopted weighted WF-SKL algorithm, while the other same dataset was processed with the algorithms without weighting. The results of the experiments are shown in Fig.6:
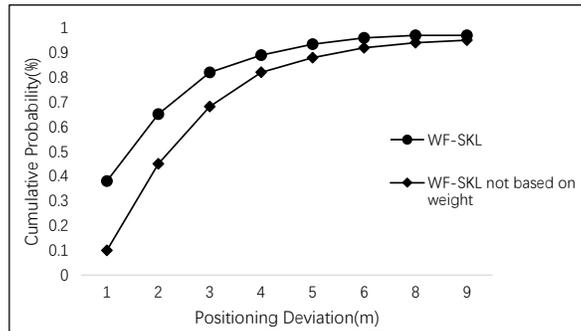

Fig. 6. Weight Influence

In Fig.6, the accuracy of non-weighted WF-SKL algorithm is much lower to estimate the position, however, the accuracy is improved a lot after introducing the weighting. Weighted value can reflect the distance of the matching points, however the average value cannot reflect the distance, therefore the accuracy of non-weighted algorithm is lower.

4.4 Scalability

In the last experiment, we compared different KNN query algorithms in the same environment. The KNN based on clusters is an improved KNN algorithm by introducing clustering and weighting, in other words, KNN based on clusters algorithm is a kind of KNN query algorithm using Euclidean distance. The performance is shown in Fig.7.
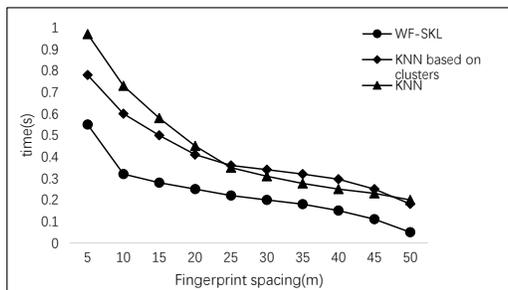

Fig. 7. Parallel Performance

As shown in Fig.7, the speed of the WF-SKL algorithm is faster than others. K-AP(P,Q) algorithm is the R*-tree based KNN query method, which could be accelerated to process big data query based on MapReduce framework. Therefore the processing speed of the paralled algorithm is much faster than other previous algorithms.

Finally, we compared the performances of the three positioning methods in Table 1. The parameters include mean error, error variance, and the error within $2M$. The accuracy of WF-SKL algorithm is in 2.35 meters. It is increased 8.2% compared with KNN algorithm, and also increased 16.96% compared with cluster-based KNN algorithm. For the WF-SKL algorithm, the rate of the error within 2 meters reaches 47.28%, meanwhile it has the lowest error variance, which means the data positioning of our method is much more reliable.

Table 1. Comparison Among Different Methods

Methods	Mean Error(m)	Error Variance(m)	Error within 2m
WF-SKL	2.35	1.48	47.28%
KNN based on clusters	2.56	1.62	38.74%
KNN	2.83	1.89	32.56%

5 Conclusion and Future Work

Currently the wireless indoor positioning is widely implemented and deployed. In this paper, we mainly focus on introducing the method of building AP fingerprint

library based on the correlations of RSSI. Furthermore, we introduce an *AP* fingerprint based WF-SKL algorithm, which adapts the KNN algorithm to a weighted K-AP query method. The algorithm eliminates several weaknesses of the old methods and improves the performance and accuracy of IPS.

References

References

1. A. Aboodi and T.-C. Wan, "Evaluation of wifi-based indoor (wbi) positioning algorithm," in *Mobile, Ubiquitous, and Intelligent Computing (MUSIC), 2012 Third FTRA International Conference on.* IEEE, 2012, pp. 260–264.
2. M. E. Rida, F. Liu, Y. Jadi, A. A. A. Algawhari, and A. Askourih, "Indoor location position based on bluetooth signal strength," in *Information Science and Control Engineering (ICISCE), 2015 2nd International Conference on.* IEEE, 2015, pp. 769–773.
3. R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The active badge location system," *ACM Transactions on Information Systems (TOIS)*, vol. 10, no. 1, pp. 91–102, 1992.
4. N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proceedings of the 6th annual international conference on Mobile computing and networking.* ACM, 2000, pp. 32–43.
5. Y. Hou, G. Sum, and B. Fan, "The indoor wireless location technology research based on wifi," in *Natural Computation (ICNC), 2014 10th International Conference on.* IEEE, 2014, pp. 1044–1049.
6. H. Zou, Y. Luo, X. Lu, H. Jiang, and L. Xie, "A mutual information based online access point selection strategy for wifi indoor localization," in *Automation Science and Engineering (CASE), 2015 IEEE International Conference on.* IEEE, 2015, pp. 180–185.
7. H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, no. 6, pp. 1067–1080, 2007.
8. C. Yang and H.-R. Shao, "Wifi-based indoor positioning," *Communications Magazine, IEEE*, vol. 53, no. 3, pp. 150–157, 2015.
9. S. Sen, R. R. Choudhury, B. Radunovic, and T. Minka, "Precise indoor localization using phy layer information," in *Proceedings of the 10th ACM Workshop on hot topics in networks.* ACM, 2011, p. 18.
10. Y. Fang, Z. Deng, C. Xue, J. Jiao, H. Zeng, R. Zheng, and S. Lu, "Application of an improved k nearest neighbor algorithm in wifi indoor positioning," in *China Satellite Navigation Conference (CSNC) 2015 Proceedings: Volume III.* Springer, 2015, pp. 517–524.
11. J. Jun, S. Chakraborty, L. He, Y. Gu, and D. P. Agrawal, "Robust and undemanding wifi-fingerprint based indoor localization with independent access points," 2015.
12. S. Yang, P. Dessai, M. Verma, and M. Gerla, "Freeloc: Calibration-free crowdsourced indoor localization," in *INFOCOM, 2013 Proceedings IEEE.* IEEE, 2013, pp. 2481–2489.
13. Y. Liu, N. Jing, L. Chen, and H. Chen, "Parallel bulk-loading of spatial data with mapreduce: An r-tree case," *Wuhan University Journal of Natural Sciences*, vol. 16, no. 6, pp. 513–519, 2011.
14. J. X.-l. PENG Dun-lu, "Algorithm for k-closest pair query based on two sets on mapreduce framework," *Journal of Chinese Computer Systems*, vol. 37, no. 3, p. 483, 2016. [Online]. Available: http://xwxt.sict.ac.cn/EN/abstract/article$_3$302.*shtml*