An Optimized Positioning Algorithm Based on Improved Gaussian Filtering

Min Chen¹, Dongmei Zhang¹, Yue Zhao¹, Taojiang Wu²*

¹ School of Electronic Information and Communication Engineering, Chongqing Aerospace Technical College, Chongqing 400021, P.R.China
²Chongqing Energy College, Chongqing 402260, P.R.China
*Corresponding author’s e-mail: taojiangfree@163.com

Abstract: WiFi offers simple, convenient, ubiquitous, and economic solutions for indoor positioning services, by matching a pre-established WiFi’s RSSI fingerprint database to a mobile terminal’s received RSSI values. A setback of this fingerprint matching method is its low precision, only miserably on order of meters, due to signal impairment by indoor complicated environment. To circumvent this, we revise the traditional weighted K-neighborhood algorithm by incorporating a Bayesian probability optimization. The proposed combination of Bayesian with weighted K-neighborhood algorithm improves the accuracy and reduces the average running time. Computer simulation shows that proposed Bayesian probabilistic optimization algorithm improves the positioning accuracy from 34% to 46%, with an average of about 14.86%, and the computation stability is also enhanced.

1. Introduction
The fast advancement of wireless communication and mobile internet have given birth to increasingly ubiquitous smart terminals and a plethora of content-oriented services. As a result, positioning technology has been put into high demand.

In the interior of a building, however, the blockage of civil constructs make the popular global positioning system simple unavailable. Instead, other radio-based positioning technologies have to be relied, such as Ultra Wide Band (UWB), WiFi, Bluetooth and Radio Frequency Identification (RFID).

UWB radio coverage is too small, location cost is high and intelligent terminals do not support this technology for the time being; Bluetooth positioning technology needs to pre-configure a large number of beacon nodes, vulnerable to noise signals and short transmission distance; Radio Frequency Identification technology has short action distance, poor anti-interference and high positioning cost; Infrared positioning technology is vulnerable to environmental interference and poor positioning effect. In contrast, WiFi positioning technology is undoubtedly the simplest, fast, convenient and economical indoor positioning technology because it does not need to add additional base stations and has a wide range of coverage

Because of the complex civil constructure, the blockage of partitioning materials, and the present of personnel, an office environment is notorious for multipath and none line of sight, detrimental to the positioning accuracy of radio-based technologies.

Some simple location algorithm such as Received Signal Strength Indication (RSSI)[1], is pre-processed by off-line phase, and a fingerprint library of WiFi signal strength is established as sample spaces. When a mobile device enter the arena, it measures the real-time signal strength and
feed the measured online to the access point, where the real-time signal is matched to the fingerprint library, so as to obtain the mobile terminal current position. Generally speaking, there are three basic principles underlying indoor positioning technologies: trilateral measurement, triangulation measurement, and signal intensity measurement. Considering the extreme complexity of indoor building and partitioning structures, and non-line-of-sight propagation and multi-path fading problem, the measured time and angle are greatly deviated from real values, and WiFi fingerprint matching suffers from low positioning accuracy. The main reason is that besides the factors mentioned above, such as human occlusion, non-line-of-sight and multi-path effects, there are also more fingerprint information stored in off-line fingerprint inventory when the location space for matching is large. Increasing the computational complexity will affect the positioning accuracy.

In order to strike a balance of location accuracy with operation time, this paper proposes a Bayesian optimization algorithm which combines Bayesian estimation with weighted K-neighborhood algorithm. It can improve the location accuracy and reduce the operation time accordingly.

2. RSSI Noise Reduction Processing
The overall RSSI sample data shows a Gaussian distribution trend, and the probability value gradually decreases when it extends to both sides. In order to better fit the distribution curve of the RSSI sample data, this paper introduces a Gaussian distribution model, assuming that the RSSI sample data obeys \((u, s^2)\) Gaussian distribution, and its probability density function formula is shown in equation (1).

\[
f(rssi) = \frac{1}{s \sqrt{2\pi}} e^{\frac{(rssi-u)^2}{2s^2}}
\]

Among them

\[
u = \frac{1}{N} \sum_{i=1}^{N} rssi_i
\]

\[
s = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (rssi_i - \frac{1}{N} \sum_{i=1}^{N} rssi_i)^2}
\]

In the formula, \(u\) is the mean value, \(s\) is the standard deviation, \(N\) is the number of RSSI sample data, and \(rssi_i\) is the RSSI sample data collected for the \(i\)-th time.

For the small probability events with large fluctuations caused by noise interference, the use of Gaussian filtering to filter out low-probability events may also lead to the loss of effective information. Therefore, this article improves the Gaussian filtering[5]. The specific process is as follows:

(1) Gaussian filtering: The Gaussian model is introduced to filter the RSSI sample data collected in the first step. This paper selects a high probability interval within 90% of the probability value, namely \(RSSI \in (u-1.65s, u+1.65s)\), Where \(u\) is the mean and \(s\) is the standard deviation.

(2) Low probability RSSI sample data: remove low probability points through the second step of Gaussian filtering, set the small probability sample set as \(T\) and the number of samples as \(k\), \(T=[RSSI_{out1}, RSSI_{out2},..., RSSI_{outk}]\). Assign values to the RSSI sample data in the set \(T\) as shown in equation (4).

\[
RSSI_{outj} = \frac{1}{N-k} \sum_{i=1}^{N-k} RSSI_i, j = 1, 2, ..., k
\]

In the formula, \(RSSI_{outj}\) is the sample data in the sample set \(T\), and \(N\) is the total RSSI sample data.

(3) Output RSSI noise reduction data: assign low probability sample data to high probability sample data, that is, complete the RSSI noise reduction process of the current fingerprint point, and save the RSSI data to the offline fingerprint database.

Comparing the two methods of smoothing mean filtering and Gaussian filtering, Figure 1 is a comparison diagram before and after RSSI filtering.
3. Matching Optimization Algorithm

3.1 Weight K-Nearest Neighborhood

The conventional Weight K-Nearest Neighborhood[3] (WKNN) is a modification to the K-Nearest Neighborhood[4](KNN). Considering that the distances of the selected K fingerprints to the point to be measured is obviously different, it does not suffice to just average the distance values in order to obtain the position of the point to be measured. As a result, WKNN uses the distance values $D_i$ ($i=1,2,...K$), between the $K$ fingerprints and the point to be measured to determine a set of weight coefficients $w_i$ as follows:

$$w_i = \frac{1}{D_i} \sum_{i=1}^{K} \frac{1}{D_i}$$

(5)

The estimated position of the point is calculated as the summation of the weight coefficients, namely

$$\hat{(x, y)} = \sum_{i=1}^{K} w_i (x_i, y_i)$$

(6)

In view of the low accuracy of WiFi fingerprint matching using traditional WKNN algorithm, this paper improves the weight coefficient[6] by introducing two more weight coefficients $w_{i1}$ and $w_{i2}$ as follows:

$$w_{i1} = \frac{D_{i+1}}{D_i + D_{i+1}}$$

(7)

$$w_{i2} = \frac{D_i}{D_i + D_{i+1}}$$

(8)

And the estimated position coordinates of the point is obtained as

$$\hat{(x', y')} = w_{i1} (x_i, y_i) + w_{i2} (x_{i+1}, y_{i+1})$$

(9)

Where $D_i$ is the distance between the $i$th fingerprint point and the point to be measured, and $D_{i+1}$ is the distance between the $(i+1)$th fingerprint point and the point to be measured.

3.2 Bayesian Probability Algorithms

The $i$th reference point (RP) in the fingerprint database receives the RSSI of the $n$th access point (AP), denoted as $S_i$. As a result, the fingerprint database can be constructed as $S=[S_1, S_2, ... , S_n]$, $j=1,...,n$,where the location information for the $i$th reference points is denoted as $L_i=(x_i, y_i)$. The posterior probability, namely, $P(L_i|S)$, of an observed signal which appears at the $i$th reference point is calculated by the following formula[2]:

![Figure 1. Comparison chart before and after RSSI filtering](image-url)
\[ P(L_i / S) = \frac{P(S / L_i)P(L_i)}{P(S)} = \frac{P(S / L_i)P(L_i)}{\sum_{i=1}^{m} P(S / L_i)P(L_i)} \]  \hspace{1cm} (10)

Where \( P(S / L_i) \) is the conditional probability of the fingerprint database given that the observed signal coincides with the \( i \)th reference point; \( P(L_i) \) is the priori probability of the \( i \)th reference point. It is generally considered that the probability, which is generally considered as obeying uniform distribution law, namely, \( P(L_i)=1/m \). Assuming that the measured signals emitted by the number of \( N \) access points are mutually independent, then:

\[ P(S / L_i) = P(s_1 / L_i)P(s_2 / L_i)P(s_n / L_i) \]  \hspace{1cm} (11)

From the related theory of probability statistics, it can be concluded that the measured signal \( s_j \) of the \( j \)th access point received by any \( i \)th to-be-tested point satisfies the Gaussian normal distribution. By selecting any one-to-be-tested point and accessing it in the experimental environment Point to perform 100 tests to verify, as shown in Figure 2.

![Figure 2](image-url)

Figure 2. Statistics of the measured signal strength when the distance between the point to be measured and the access point is 3m where \( u \) is the measured signal of the \( j \)th access point received by the \( i \)th reference point, namely:

\[ P(s_j / L_i) = \frac{1}{\sqrt{2\pi} \sigma} \exp\left[ -\frac{(s_j - \mu)^2}{2\sigma^2} \right] \]  \hspace{1cm} (12)

Finally, using \( P(L_i/S) \) as the weight coefficients for the \( L_i \) position information corresponding to the \( i \)th reference point, we obtain the estimated position as follows:

\[ \hat{x}, \hat{y} = \sum_{i=1}^{m} P(L_i / S)(x_i, y_i) \]  \hspace{1cm} (13)

The flow of Bayesian probability algorithm is as follows:

**Input:** The number of \( M \) APs’ measured signal \( S \) and position information \((x, y)\); the signal \( S_k \) observed by a mobile terminal.

**Output:** Final location information of the terminal \( \hat{x}, \hat{y} \).

For \( i=1 \) to \( m \):

1. calculate \( P(S/L_i) \);
2. calculate \( P(L_i/S) \);
3. obtain \( \hat{x}, \hat{y} = \sum_{i=1}^{m} P(L_i / S)(x_i, y_i) \)

End
3.3 Bayesian Optimal Algorithms

Since the afore-mentioned Bayesian probability algorithm needs to calculate the posterior probability corresponding to every reference point when estimating a single mobile position, a large amount of computation is required, which leads to great latency. To solve this problem, we propose a Bayesian probability optimization algorithm by integrating WKNN into Bayesian probability algorithm, in order to lessen computational burden and improve localization accuracy.

In the off-line database-building stage, we choose \( M \) reference points, where the RSSI signals from \( N \) APs are measured and built into a set \( S \), namely \( \{S_1, S_2, \ldots, S_m\}^T \). The \( i \)th reference-point’s data set is \( \{(S_{i1}, S_{i2}, \ldots, S_{im}), (x_i, y_i)\} \), where \( (S_{i1}, S_{i2}, \ldots, S_{im}) \) denotes the measured signal set \( S \), and \( (x_i, y_i) \) denotes the current position \( L_i \) of the \( i \)th RP.

In the position-matching stage, the \( k \)th to-be-measured position, with RSSI data recorded as \( S_k = \{S_{k1}, \ldots, S_{kj}, \ldots, S_{kn}\} \), \( k \in N^+ \), is to have its distances to any \( j \)th reference point in the fingerprint database as follows:

\[
D_i = \left( \sum_{j=1}^{n} |S_{ij} - S_{ij}|^q \right)^{1/q}, \quad i = 1, 2, \ldots, m
\]  

(14)

Where \( q=1 \) represents absolute distance; \( q=2 \), the Euclidean. Here, we choose \( q=2 \). The smaller the value of \( D_i \), the greater the similarity between the two points.

Using Euclidean distance, we first calculate the \( T \) numbers of \( D_i's \), \( D_i = \{D_{i1}, \ldots, D_{in}, D_{i} \} \), sort them by ascending order, \( \{(S_{i1}, S_{i2}, \ldots, S_{im}), (x_i, y_i)\}, i < l+1 \), and then using WKNN to select the first \((t-1)\)'s nearest neighbors, \( \{(S_{i1}, S_{i2}, \ldots, S_{im}), (x_i, y_i)\}, i < l+1 \). We further use Bayesian probability algorithm to calculate the final position coordinates. The flowchart of Bayesian probability optimization algorithm is shown in Figure 3 and stated in text as follows.

Algorithm:

Input: The number of \( m \) APs’ measured signal \( S \) and position information \( (x, y) \); the signal \( S_k \) observed by a mobile terminal.

Output: Location coordinates of the terminal, \( \hat{x}, \hat{y} \).

For \( i = 1: m \):

1. Calculate the Euclidean distances between the terminal point and Each RP;
2. Sort the \( T \)’s Euclidean distances \( D_i’s \) in ascending order;
3. Using WKNN algorithm to select the first \((t-1)\)'s \( L_k \).

End

For \( i = 1: m \):

1. Calculate the first \((t-1)\)'s \( P(S|L_i) \);
2. Calculate the first \((t-1)\)'s weight coefficients \( P(L_i/S) \);
3. Obtain the final position coordinates \( \hat{x}, \hat{y} \).

End

4. Experimental results and analysis

4.1 Layout of experimental environment

For experiment environment, we choose a typical office with a floor area of \( 14 \text{m} \times 8 \text{m} \). This experiment uses the TP-LINK wireless router as the AP access point, and the Huawei EMUI 9.1.0 mobile phone terminal as the RP reference point collector and the test point for subsequent testing; in order to ensure that the RP reference point collected in the experimental area comes from each AP access point. The RSSI sequence of the entry point is different. The experiment tested the signal "distance-loss" of each
AP node. As shown in Figure 3 below, the measured signal strength value of the AP node's distance in the range of 7-10m tends to be flat, but combined with the actual cost For problems, the arrangement can be within 7m.

Figure 3. AP access point "distance-loss" line graph

Divide the experimental area into a number of square grids according to the 0.65m 1m specification. Use each square grid vertex as the RP reference point and mark the serial number from left to right and from bottom to top. A total of 132 RP reference points are marked, and Collected ten times in different directions at each RP reference point. The fingerprint library has 1320 fingerprints. The AP access point layout is shown in Figure 4 below.

Figure 4.14m×8m experimental area map

4.2 Experimental Result
In order to verify the superiority of the Bayesian probability optimization algorithm proposed in this article, in the experimental area, the laboratory selects two points to be measured \((tx_1, ty_1)\) and \((tx_2, ty_2)\), and each point is measured when the equipment is in a static state 15 sets of positioning results, and record the real coordinates and the positioning coordinates under the traditional WKNN algorithm, Bayesian probability, and Bayesian probability optimization algorithm. The test results are shown in Figure 5 and Figure 6.
As shown in Figure 5 and Figure 6, in a static state, the proposed Bayesian probability optimization algorithm has a closer estimate to the real position coordinates than the WKNN and Bayesian probability algorithms do, and has a smaller deviation, and stronger stability.

4.3 Experimental Analysis

To account for the average positioning error, we cumulatively measure \( k \) points in the experimental area, and obtain average positioning error \( \bar{d}_{\text{error}} \) as follows:

\[
\bar{d}_{\text{error}} = \frac{\sum_{i=1}^{k} \sqrt{(x_i - tx_i)^2 + (y_i - ty_i)^2}}{k}
\]  

In particular, we let \( k = 100 \), namely, select 100 independent and uncorrelated test points in the experimental area, and use the conventional WNNN, Bayesian probability, and Bayesian probability optimization algorithms respectively to estimate and position coordinates, and compare the estimates with the real coordinates. Figure 7 shows the resultant Cumulative Distribution Function (CDF) of the estimation errors.
As shown in Figure 7, the positioning accuracy of the conventional WKNN algorithm can reach 34% within error of 1 meter; in contrast, the positioning accuracy of Bayesian probability optimization algorithm can reach 46%. Moreover, the positioning accuracy of the conventional WKNN algorithm can reach 69% within error 2 meters, and that of Bayesian probability optimization algorithm can reach 76%. The improvement of latter is obvious. Table 1 below shows the comparison of positioning accuracy and real-time performance of the algorithm.

| Algorithm                  | Ave. error /m | Standard devi. /m | Average running time /s |
|----------------------------|---------------|-------------------|-------------------------|
| WKNN                       | 1.6610        | 1.4704            | 0.20361                 |
| Bayesian                   | 1.5982        | 1.3641            | 0.53151                 |
| Bayesian Optimization      | 1.4142        | 1.4265            | 0.20470                 |

As evident in Table 1, the average positioning error of the conventional WKNN algorithm is 1.6610m; that of Bayesian probability algorithm is 1.5982; and that of our proposed Bayesian probability optimization is 1.4142. In general, our proposed algorithm has improved the positioning accuracy by 14.86% and 11.51% respectively. In terms of average running time, our proposed Bayesian probability optimization algorithm reduces the average running time by about 2.60 times in comparison with the conventional Bayesian probability algorithm. In terms of positioning stability, our proposed Bayesian probability optimization algorithm is more stable than the conventional WKNN algorithm. In conclusion, the proposed Bayesian probabilistic optimization algorithm has certain improvement in positioning accuracy, stability and real-time performance compared with the existing algorithms.

5. Conclusion
In this paper, we propose a Bayesian probability optimization algorithm, which combines the conventional Bayesian probability algorithm with WKNN algorithm, in order to improve indoor positioning accuracy and to reduce computational cost. We validate our proposed algorithm in a real setting consisting an experimental office area and a corridor, with office equipment and office clerk present. The test results show that our proposed Bayesian probability optimization algorithm improve the positioning accuracy by about 14.86% compared with the traditional WKNN, and has a stronger operational stability. However, in this article, there is still a way to improve, such as an average running time increase of about 0.54% relative to the WKNN algorithm. The future work needs to be discussed further improve the real-time and stability of the algorithm.

Acknowledgements
The authors acknowledge funding from Key project of technological innovation and application development in chongqing(cstc2019jscx-mbdxX0002).
Reference

[1] TIAN Jiaying. Research on indoor positioning technology based on WIFI location fingerprint algorithm [D]. Lanzhou: lanzhou jiaotong university, 2018:18-22.

[2] RUAN Chao, FENG Tao, GUO Kaixuan. Indoor WiFi location algorithm based on pca-wbayes[J]. Sensor and microsystem, 2018, 37(8):124-126.

[3] HE Zhishuang, WAN Yaping, ZHAO Qing, et al. Study on the improvement of indoor WiFi location technology [J]. Journal of south China university (natural science edition), 2018, 32(3):66-70.

[4] Minh Tu Hoang, Yizhou Zhu, Brosnan Yuen, et al. A Soft Range Limited K-Nearest Neighbors Algorithm for Indoor Localization Enhancement[J]. IEEE Sensors Journal, 2018, 18(24):10208-10216.

[5] Oh J , Kim J , Adaptive K-nearest neighbour algorithm for WiFi fingerprint positioning[J], ICT Express, 2018,4(2):91-94.

[6] C. Zeng, S. Zhao, Y. Zhong, et al. An Improved Method for Indoor Positioning of Wifi Based on Location Fingerprint[C] // 2018 7th International Conference on Digital Home (ICDH). Guilin: IEEE, 2018:280-285.

[7] H. X. Jian, W. Hao. WIFI Indoor Location Optimization Method Based on Position Fingerprint Algorithm[C] // 2017 International Conference on Smart Grid and Electrical Automation (ICSGEA). Changsha: IEEE, 2017:585-588.