A Fingerprint Positioning Algorithm Based on Multilevel Manual Definition

Xiao-zhi DU*, Xiao-song LI and Xue WANG

School of software, Xi'an Jiaotong University, Xi'an, 710049, China
*Corresponding author

Keywords: Indoor positioning, Fingerprint database optimization, Multilevel matching.

Abstract. In recent years, more and more attention has been drawn to the indoor positioning algorithm based on fingerprint model. Although, such algorithm is irrelevant to the location of the signal source, it has the drawbacks of low accuracy and poor real-time capability. This paper proposes a fingerprint positioning algorithm based on multilevel manual definition. In the offline phase of the fingerprint positioning algorithm, source data in the fingerprint database is processed by average weighting method, and a high precision fingerprint database is established to improve matching accuracy. In the online phase, multilevel matching algorithm is employed to shorten matching time. Experiments based on ZigBee technology were conducted, and the results showed that the algorithm for establishing a fingerprint database can improve matching precision by 50% and shorten matching time by 50% compared to the mode of RSSI.

Introduction

Recently, wireless sensor network technology and intelligent home are increasingly widespread, resulting in a significance of indoor position. The indoor positioning system mainly includes ultrasonic positioning technology, infrared positioning technology, and the received signal strength indication (RSSI) positioning technology. Wireless signal positioning technology can be classified into two types, namely range-based and range-free. The range-based positioning makes use of the propagation characteristics of signals. According to the nonlinear attenuation properties of the wireless signal, signal strength can be converted to distance by channel model, and then the position of the user is inferred by trilateration. Fingerprint model positioning is based on a large number of actual observed data. After trained by machine learning, the positioning model can be used to estimate the position of the target point.

In the implementation, location fingerprint can be divided into two stages. The first stage is data acquisition, or offline phase; the second stage is matching positioning, or online phase. In the offline phase, the location fingerprints of each reference node in the target area are recorded, such as the position coordinates of the reference node and the received signal strength. Every reference node corresponds to a specific set of location fingerprints, thereby forming a location fingerprint database. In the online phase, the observer uses the characteristics of current position of the target object to match the location fingerprint in the database by matching algorithm in order to determine the location of the target object. This paper mainly discusses how to establish high precision offline fingerprint database and how to improve fingerprint database matching speed.

Fingerprint Positioning Algorithm Based on Multilevel Manual Definition

Offline Fingerprint Database Optimization

Establishing fingerprint database is the basis for the matching in the real-time phase. The original location fingerprint (used to establish fingerprint database) collected in the offline phase ought to be “zero distance” matched (which means, in the offline phase, the location of the fingerprint used to establish the fingerprint database is the same as the original location; zero matched rate = number of zero matched points / number of total sample points). However, due to the influence from various factors, there might be a deviation as Table 1 shows. To avoid these deviations, many previous
studies often simply used the mean or the mode of RSSI, or even assumed that the RSSI accorded with probability distribution. However, multiple RSSIs instead of one in one location, are needed to synthesize into one fingerprint. This paper takes the following data processing method to improve the accuracy of fingerprint database.

Table 1. Use the Mode of Location Fingerprint to Establish a Fingerprint Database by Nearest Neighbor Algorithm

<table>
<thead>
<tr>
<th>Actual Position</th>
<th>Occurrence Number</th>
<th>Position 1</th>
<th>Occurrence Number</th>
<th>Position 2</th>
<th>Occurrence Number</th>
<th>Position 3</th>
<th>Occurrence Number</th>
<th>Position 4</th>
<th>Occurrence Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>(11,5)</td>
<td>206</td>
<td>(11,5)</td>
<td>3</td>
<td>(12,3)</td>
<td>1</td>
<td>(16,2)</td>
<td>14</td>
<td>(17,6)</td>
<td>188</td>
</tr>
<tr>
<td>(14,5)</td>
<td>267</td>
<td>(13,4)</td>
<td>2</td>
<td>(14,1)</td>
<td>4</td>
<td>(14,5)</td>
<td>2</td>
<td>(17,2)</td>
<td>259</td>
</tr>
<tr>
<td>(15,2)</td>
<td>194</td>
<td>(15,2)</td>
<td>9</td>
<td>(19,1)</td>
<td>94</td>
<td>(22,2)</td>
<td>1</td>
<td>(23,2)</td>
<td>1</td>
</tr>
<tr>
<td>(23,1)</td>
<td>204</td>
<td>(23,2)</td>
<td>111</td>
<td>(24,2)</td>
<td>3</td>
<td>(23,1)</td>
<td>186</td>
<td>(25,2)</td>
<td>4</td>
</tr>
<tr>
<td>(5,2)</td>
<td>282</td>
<td>(3,1)</td>
<td>1</td>
<td>(4,4)</td>
<td>271</td>
<td>(5,2)</td>
<td>9</td>
<td>(5,6)</td>
<td>1</td>
</tr>
<tr>
<td>(5,5)</td>
<td>202</td>
<td>(3,6)</td>
<td>1</td>
<td>(9,1)</td>
<td>119</td>
<td>(10,3)</td>
<td>10</td>
<td>(10,4)</td>
<td>72</td>
</tr>
<tr>
<td>(6,1)</td>
<td>227</td>
<td>(3,2)</td>
<td>21</td>
<td>(6,1)</td>
<td>6</td>
<td>(6,2)</td>
<td>184</td>
<td>(12,1)</td>
<td>16</td>
</tr>
</tbody>
</table>

Using the mode of RSSI to establish a fingerprint database, select the data which is collected from a data collection point for N times (assume this point as A, and the coordinates of this point is \((x_A, y_A)\) to analyze, and then match each collected point by nearest neighbor algorithm to get a matching position \(B_i\), with the coordinates \((x_{B_i}, y_{B_i})\), then using the Eq. 1 to compute the distance between \(B_i\) and A.

\[
d_i = \sqrt{(x_A - x_{B_i})^2 + (y_A - y_{B_i})^2}
\]  

(1)

Where \(d_i\) represents the distance between \(B_i\) and A. Then weighted by distance, when \(d_i = 0\), set its weight \(\omega = 1/k\), where k denotes the frequency of \(d_i = 0\)

\[
RSSI_1 = \sum_{j=0}^{k} RSSI_j / \sqrt{k}
\]  

(2)

When \(d_i = 0\), \(RSSI_j\) is the location fingerprint of the sample. When \(d_i \neq 0\), set its weight value \(\omega = 1/d_i\), and assume

\[
RSSI_2 = \left(\sum_{j=0}^{n-k} RSSI_j / d_j\right) / \left(\sum_{j=0}^{n-k} 1 / d_j\right)
\]  

(3)

Where \(RSSI_j\) is the location fingerprint of the sample when \(d_i \neq 0\).

Finally, the location fingerprint vector used to establish fingerprint database is

\[
RSSI = \left\{ \begin{array}{ll}
RSSI_1, & \text{if } d_i = 0, \\
\alpha \times RSSI_1 + (1 - \alpha) \times RSSI_2, & \text{if } d_i \neq 0, \\
\end{array} \right. \text{, (} 0 < k < n, \alpha \in (0, 5, 1) \right)
\]  

(4)

After computing every sample point, the location fingerprint database can be established.

**Multilevel Matching**

This algorithm classifies points by some simple clustering algorithm, and eliminates the crossover artificially. There is obvious class boundary in the classification of the location fingerprint. The membership of the location fingerprint in this class is 1, but 0 to other classes. However, because of various factors in practical, the data to be processed is not accurate. The mode feature of such data is ambiguous in behavior and class attribute. For example, in location fingerprint classification, the mode feature of the location fingerprint on the classification boundary is ambiguous. These location
fingerprints contain the mode features of the neighboring classes at the same time. As a result, it is improper to sort them into one class. The location fingerprint generated in the real-time positioning phase may not be matched to its most similar location fingerprint which may influence the final location estimation results. Thus, this algorithm refers to soft division method to eliminate fuzzy boundary. In the real-time matching phase, classes with the first and the second smallest distance are taken as matching class.

Because two classes are taken as matching class, the number of match is still large. As a result, multilevel classification algorithm is adopted to reduce the number of matching. Taking multilevel classification in the offline phase, the fingerprint characteristic value representing each class is stored in the database (e.g. the mean of each class). In the real-time phase, compute the distance between the location fingerprint of the unknown point and the fingerprint characteristic value of each classification, find the minimum as well as the second minimum value and compute next stage classification, and then repeat the above procedures until a satisfactory result is achieved.

\[
\omega_j = \frac{(1/d_j)}{\sum_{j=1}^{k} 1/d_j}
\]  

(5)

thus, the position coordinate of the unknown point is

\[
P_a = \sum_{j=1}^{k} \omega_j \times P_{aj}
\]  

(6)

\(P_a\) is the position coordinate of the unknown point, \(\omega_j\) is the \(j^{th}\) fingerprint weight value, \(P_{aj}\) is the \(j^{th}\) fingerprint.

Figure 1. Store classified information by stage in offline phase

Figure 2. Matching by stage in online phase

In the offline phase, the storing information includes the mean of the classification of each level and the points in the last level. In the real-time phase, considering the complicated indoor environment, measurement error still exists. Weighted nearest neighbor method is employed to solve this problem. The weighted factor \(1/d_j\) is used to compute the coordinates of the Euclidean distance of the location fingerprint, then weights and averages. The closer the fingerprint information point is, the higher the weight value will be obtained. Its weight value is:

\[
\frac{1}{\sum_{j=1}^{k} \frac{1}{d_j}}
\]  

Figure 1. Store classified information by stage in offline phase

Figure 2. Matching by stage in online phase

In the offline phase, the storing information includes the mean of the classification of each level and the points in the last level. In the real-time phase, considering the complicated indoor environment, measurement error still exists. Weighted nearest neighbor method is employed to solve this problem. The weighted factor \(1/d_j\) is used to compute the coordinates of the Euclidean distance of the location fingerprint, then weights and averages. The closer the fingerprint information point is, the higher the weight value will be obtained. Its weight value is:

\[
\omega_j = \frac{(1/d_j)}{\sum_{j=1}^{k} 1/d_j}
\]  

(5)

thus, the position coordinate of the unknown point is

\[
P_a = \sum_{j=1}^{k} \omega_j \times P_{aj}
\]  

(6)

\(P_a\) is the position coordinate of the unknown point, \(\omega_j\) is the \(j^{th}\) fingerprint weight value, \(P_{aj}\) is the \(j^{th}\) fingerprint.
Experimental Results and Evaluation

Experimental Environment

The experimental room was a 2555cm × 685.5cm hall with a 498.5cm × 347cm small room in it, and the indoor facilities were shown in Figure 4 (a). There were four sofas in the hall (the back of sofas were about 0.8m high), two coffee tables (about 0.4m).

![Figure 3. Experimental ichnography](image)

Figure 3. Experimental ichnography

Figure 3. (b) was the simplified graph of this room. The points in the figure were the location nodes. The height of the nodes on the wall was 1.8m and the distance of the both sides of the “1” node was 342.75cm (equally divided) whereas the distance among the nodes “2”, “3”, “4” on the right side of the wall was 638.75cm (equally divided). Similarly, the distance of the both sides of the “5” node was 349.25cm and the “6” node was 173.5cm. The distance between the “7” and “8” node was 685.5cm. The “9” node was placed on the desktop in the middle of the room (about 1m high). On the basis of Figure 3. (a), the facilities were removed which impose no influence on the location and then the user-unreachable zones were determined. Therefore the distribution of sampling points in the room was as Figure 3 (c). In the area that user could reach, the spacing of 0.8m×0.8m uniform distribution of sampling points were selected which located at the intersection of lines. Their labels are the combination of their position coordinates (lower right corner is 11, top left corner is 303; single-digit is the abscissa).

Nine positioning nodes and a target node were arranged at suitable location shown as Figure 3 (b). The target node was set to broadcast a positioning signal every 2.5s. The routing node hanging on the wall received the signals sent by the target node to obtain corresponding signal strength, then the signals were forwarded to the coordinator node in the middle. The coordinator node was used for networking and collecting messages forwarded by the destination node and the routing nodes.

Experimental results

In this experiment, 70 points were selected in the lab. According to the Eq. 4, different values of α were adopted, and different levels as well as k were set for the KNN algorithm to classify value. And the comparative results are as follows:

<table>
<thead>
<tr>
<th></th>
<th>Maximum error</th>
<th>Minimum error</th>
<th>Average error</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>6.340m</td>
<td>0.076m</td>
<td>1.449m</td>
</tr>
<tr>
<td>0.7</td>
<td>5.646m</td>
<td>0.077m</td>
<td>1.399m</td>
</tr>
<tr>
<td>0.9</td>
<td>4.756m</td>
<td>0.060m</td>
<td>1.386m</td>
</tr>
<tr>
<td>Filtering algorithm</td>
<td>4.956m</td>
<td>0.075m</td>
<td>1.756m</td>
</tr>
<tr>
<td>Unoptimized</td>
<td>10.472m</td>
<td>0.269m</td>
<td>3.967m</td>
</tr>
</tbody>
</table>
Table 3. Comparison of time consuming and accuracy when using different levels

<table>
<thead>
<tr>
<th>Level</th>
<th>Average consumed time</th>
<th>Average error</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.035s</td>
<td>1.480m</td>
</tr>
<tr>
<td>4</td>
<td>0.027s</td>
<td>1.503m</td>
</tr>
<tr>
<td>5</td>
<td>0.024s</td>
<td>1.503m</td>
</tr>
<tr>
<td>c=3 Clustering method</td>
<td>0.031s</td>
<td>1.490m</td>
</tr>
<tr>
<td>Using direct matching</td>
<td>0.054s</td>
<td>3.562m</td>
</tr>
</tbody>
</table>

Table 4. Comparison of matching accuracy when using different K parameters

<table>
<thead>
<tr>
<th>K</th>
<th>Average error</th>
<th>Maximum error</th>
<th>Minimum error</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1.546m</td>
<td>6.980m</td>
<td>0.033m</td>
</tr>
<tr>
<td>3</td>
<td>1.499m</td>
<td>6.340m</td>
<td>0.003m</td>
</tr>
<tr>
<td>4</td>
<td>1.391m</td>
<td>5.068m</td>
<td>0.077m</td>
</tr>
</tbody>
</table>

Analysis of Experimental Results

Table 2 shows that the fingerprint database optimization algorithm which reduces the average positioning error to more than 50%, and enables a higher positional accuracy within 1.5 meters. The average error tends to decrease with the increase of $\alpha$, although the effect of the variation of $\alpha$ on the results is not obvious. Therefore, the value of $\alpha$ should be as close to 1 as possible in practice. It is also found in the experiment that after optimizing the fingerprint database, if the optimized 0 matched rate is still very low, the algorithm can be reused.

Table 3 shows that the average matching time is obviously shortened. At the same time, it can be found that the more level is classified, the less time is consumed, though without a linear relation. The main reason is that more levels induced the increase of the number of query database and the ranking times, thus the decreasing speed of time consumption slows down.

Table 4 shows that the average error tends to decrease with the increase of K when the K-Nearest Neighbor matching algorithm is used.

Summary

This paper presents a fingerprint positioning algorithm based on multilevel manual definition. The experimental results show that using this algorithm can effectively reduce the matching time in the real-time phase and thus save the energy of mobile terminal. This algorithm improves the accuracy of positioning, but it increases the times of matching. In future, we will optimize the process of matching.

Acknowledgement

This research was financially supported by the Chinese National Natural Science Foundation (11575138), the Industrial PR project of Shaanxi Province (2013K06-20) and the Fundamental Research Funds for the Central Universities in China.

References


