Map matching algorithm in indoor localization

Yi Dong

Abstract—The report designs an improved map matching algorithm for Wifi indoor localization,which considers many kinds of factors such as distance,direction, geometry characteristic, history track,etc It also designs different weight definition and calculating methods for corresponding standard

Index Terms—Map matching, WIFI, GPS, indoor localization.

1 Introduction

With the rapid development of the mobile phone,Demand for location services is also growing significantly.Unlike the outdoor localization,indoor localization is affect by many factors like small enviroment, serious path loss.etc. So,the indoor localization system is different from outdoor indoor localization. the indoor localization scheme mainly leverages the wireless local area network (WLAN) access points (APs) to derive the location. Many efforts have been made in the past years, where the proposed techniques can be categorized into the fingerprint-based and the model-based scheme. While the localization accuracy of the model-based systems is not admirable, the fingerprint-based schemes have been attracting much attention in recent years. we introduce the map matching which is widely used in GPS system, and adjust it to the Wifi fingerprint-based indoor localization.

Hence, the purpose of this report is to discuss some simple map matching algorithms that can be used to reconcile inaccurate locational data with an inaccurate map/network.

Our objective in this paper is not to provide a definitive evaluation of different map matching algorithms. Rather, our objective is to describe some simple algorithms and to consider theoretically , how they might work in indoor enviromnet and provide a algorithm that might work.

We will firstly discuss the Wifi fingerprint method in the next section, show some map matching algorithm in the third section and provide our algorithm in fourth section.

2 Wifi indoor localization method

Wi-Fi indoor localization has been developed rapidly in recent years, Scholars have done a lot of research work in this field. Wi-Fi localization use the 802.11 wireless LAN, providing location service data communication at the same time. Wi-Fi localization has different Method,fingerprint localization is the most commonly used method. Wi-Fi fingerprint is the received signal Strength that receive from many 802.11 wireless access points , if we use the SS represent the received signal Strength, AP represent wireless access point and n represent number of wireless access point. the wifi fingerprint can be expressed as:

$$SF = ((AP_1, SS_1), (AP_2, SS_2), \cdots, (AP_n, SS_n))$$

When the actual positioning, 802.11 wireless adapter on the user of the mobile device collect the current position of the unknown fingerprint signal, positioning algorithm compare the signal fingerprint database to the fingerprint signals, the closest sample point is considered as the current location of the user, the algorithm is called nearest neighbor algorithm.

Nearest Neighbor algorithm uses different criteria to determine the nearest neighbor, Euclidean distance between the signal fingerprint is a very commonly used method.Assume that $SF_x = ((AP_{x1}, SS_{x1}), (AP_{x2}, SS_{x2})....(AP_{xn}, SS_{xn}))$ represent the current fingerprint of the unknown position x ,then Euclidean distance is determined:

$$\text{EucDis}(\text{SF}_x, \text{SF}_i) = \sqrt{\sum_{j=1}^{n} (\text{SS}_{xj} - \text{SS}_{ij})^2}$$

the current location of the user L_x is the sample point that has the minimum Euclidean distance.

$$L_x = \{L_m \mid \forall i, \text{EucDis}(\text{SF}_x, \text{SF}_m) \leqslant \text{EucDis}(\text{SF}_x, \text{SF}_i)\}$$

Nearest neighbor algorithm efficiency depends on the number of sampling points N, if using linear search, the time complexity of the algorithm is O (N). For small buildings and area, the algorithm is feasible.

3 Map matching algorithms

There are a number of different ways to do the map matching ,We will discuss several of them before moving on to a discussion of the specific algorithms that we considered.

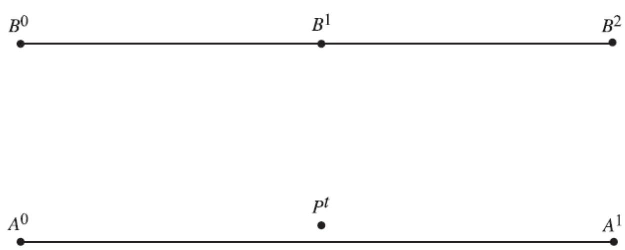

Fig.1 point to point matching

3.1 Map matching as a search problem

One can view the map matching problem as a simple search problem. Then the problem is to match P to the closest node or shape point in the network.

A number of data structures and algorithms exist (see e.g., Bentley and Maurer, 1980; Fuchs et al., 1980) for identifying all of the points near a given point (often called a range query). It is then a simple matter to find the distance between P and every node and shape point that is within a reasonable distance of it (regardless of the metric used), and select the closest.

While this approach is both reasonably easy to implement and fast, it has many problems in practice. Perhaps most importantly, it depends critically on the way in which shape points are used in the network. To see this, consider the example shown in Fig. 1. Here, P^t is much closer to B^1 than it is to either A^0 or A^1, hence it will be matched to arc B even though it is intuitively clear that it should be matched to arc A. Hence, this kind of algorithm is very sensitive to the way in which the network was digitized. That is, other things being equal, arcs with more shape points are more likely to be matched to.

One might argue that this problem could be overcome simply by including more shape points for every arc. Unfortunately, this dramatically increases the size of the network and is not guaranteed to correct the problem.

3.2 Map matching as statistical estimation

One can also view map matching as a problem of statistical estimation. In this approach, one considers a sequence of points $(P^s; ...P^t)$. and attempts to fit a curve to them. This curve is constrained to lie on the network. This kind of approach has been explored in numerous papers (see e.g., Krakiwsky et al., 1988; Scott and Drane, 1994; Jo et al., 1996) and is quite appealing. It is particularly elegant when the model describing the "physics of motion" is simple (e.g., movement is only possible along a straight line). Unfortunately, in most practical applications, the physics of motion is dictated by (or constrained by) the network. This makes it quite difficult to model. To understand why this is important, consider the network shown in Fig. 2. In this example, the positions $P^1...P^7$ have been recorded. Our objective is to fit a curve to these points, but the curve is constrained to lie on the network. In this case, there are two candidate curves, A and B (we ignore the rest of the network for simplicity). In general (i.e., regardless of the metric), the curve P is closer to the curve B than it is to the curve A.

Thus, if one uses a simple model of motion one will be led to match P to B rather than to A.

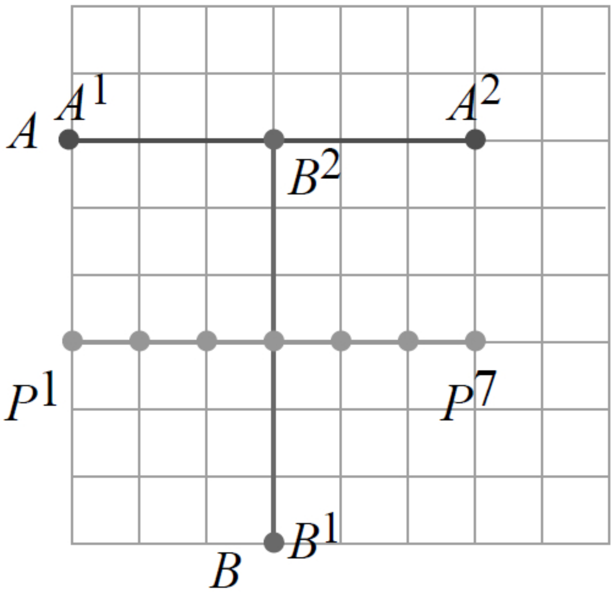

Fig.2 curve fitting

4 ALGORITHMS USED IN WIFI INDOOR LOCALIZATION

Our objective is to adjust the map matching algorithm to the Wifi environment. Algorithm 1 is very simple. but it can be quite "unstable" because it does not make use of "historical" information and direction information.

Hence, Algorithm 1 will play the role of a "straw man". It is fast, easy to implement, and should be easy to beat.

Algorithm 2 is used in GPS system.However ,it is not easy to do the curve fitting in the WIFI indoor environment.

Our algorithms is showed below,it is based on algorithm 1:

Because Wifi signal does not have the information of user's direction,so the direction information should be got from other senors on user's mobile phone,often from a digital compass.

As showed in Figure.3,P (including P1, P2, P3) is the current point to matching, Ll, L2, L3 is the Center line that near the point P. In this matching algorithm, do the projection from the unmatched point to all the route around it, calculate the projection distance $r1$ and the angle $\alpha 1$ between direction of unmatched point and the route.Remove the r1 and $\alpha 1$ that is smaller than the threshold value to get rid of some bad data.Then calculate the weight for each candidate route as:

$$W = W_\alpha + W_r$$

W_α and W_r is the weight of distance and angle.In all the candidate routes, choosing the maximum value of W as the matching path. Finally, the algorithm matches the point to the road using the projection point to display the current position of the user. In Figure 3, after the above calculation, choose L3 as the matching route and the projection points Ql of the point P on the route as the user's current position.

When dealing with the intersection of the route, according to the angle of road connectivity , choosing the route which it's angle changes smaller, as the matching results.Define W_α as:

$$W_\alpha = A_w cos(\alpha)$$

Where α is the angle between the direction of the user and the direction of the road.A_w is the expansion factor, its value determines the size of the W_α. When angle changes small,A_w is choosed small.

Define W_r as:

$$W_r = A_r tan^{-1}(\frac{1}{r})$$

where r is define as:

$$r = \frac{x_3(y_1 - y_2) - y_3(x_1 - x_2) + (x_1 y_2 - x_2 y_1)}{\sqrt{(x_1 - x_2)^2 - (y_1 - y_2)^2}}$$

Where x1,x2,x3,y1,y2,y3 is showed in fig 4.A_r is another expansion factor, its value determines the size of the W_r.

Select the maximum weight in the all the candidate routes as a matching path.Then matching the point to the projection on the selected route as the matching result.

This algorithm logic is simple, fast, real-time, required less memory space, we aslo can adjust the weights of the expansion factor to improve the accuracy in corners and intersections.

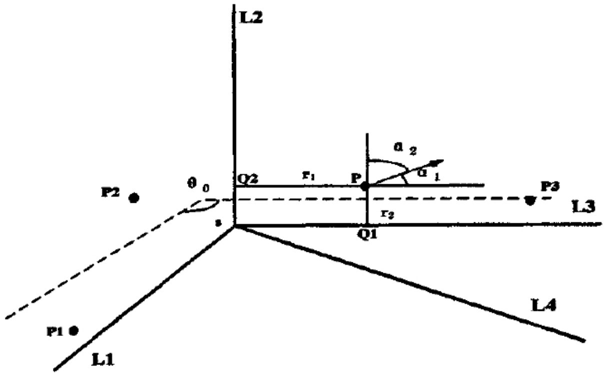

Fig.3

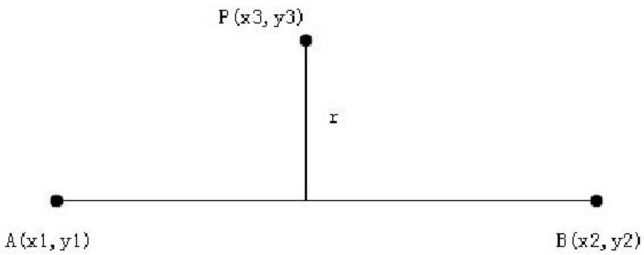

Fig.4

5 CONCLUSIONS AND FUTURE RESEARCH

In this report, we have described several algorithms (or parts of algorithms) for matching an user's position to a indoor route. Obviously, more work needs to be done.

Firstly, the 3-D condition must be taken into consideration.

Secondly,a wider variety of algorithms need to be evaluated. Particular attention needs to be given to the problems that arise at intersections. the algorithm we used is not immediately obvious that it will work best.

Finally, more attention needs to be given to how different algorithms can be compared empirically. This is a particularly thorny problem because it is quite difficult to measure the "true position" outside of a laboratory or test track. so,we should do some simulation work to this algorithm and compare it with other algorithms.

6 REFERENCES

Locating in Fingerprint Space: Wireless Indoor Localization with Little Human Intervention Zheng Yang, Chenshu Wu, and Yunhao Liu School of Software and TNList, Tsinghua University

A survey of mobile phone ensing Nicholas D. Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T. Campbell, Dartmouth College

Indoor Localization with a Crowdsourcing based Fingerprints Collecting Zhengyong Huang, Fan Wang, Hui Yu, Xiaohua Tian, Yunfeng Guan, Zhibang Ge Dept. of Electronic Engineering, Shanghai Jiao Tong University, China