

Map Matching with Hidden Markov Model on Sampled Road Network

Rudy Raymond, Tetsuro Morimura, Takayuki Osogami
IBM Research – Tokyo
{raymond,tetsuro,osogami}@jp.ibm.com

Noriaki Hirose
Kyoto University
n_hirosue@amp.i.kyoto-u.ac.jp

Abstract

This paper presents a map matching method based on an ideal Hidden Markov Model (HMM) to find a sequence of roads that corresponds to a given sequence of raw GPS points. Our method is a simplification of the more-complex HMM-based method that maintains its capabilities to cope with the noises and sparsity of the raw GPS data. We test the method with the real-world raw GPS data that is publicly available. Experiments show that despite its simplicity, the proposed method performs sufficiently well under sparse GPS points and sparse road network data.

1. Introduction

As more mobile devices with GPS (Global Positioning System) sensors are available, matching raw GPS points to roads, or *map matching*, is becoming important to recognize mobility patterns in many real-world applications. The map matching itself can be considered as a pattern recognition task, while in other cases, it is a prerequisite for advance analytics in pattern recognition with GPS sensor data.

Although applications such as car navigation require the ability to pinpoint the location of a moving object, in many sensing applications, such as recovering routes of buses [5], or discovering roads passed for telematics car insurance, a rougher estimation of identifying the sequence of roads that were passed by is often sufficient. For this purpose, map matching technologies on low GPS sampling rate are required [7]. Nevertheless, they are hindered by the sparsity of the GPS data and errors that can come from the measurement devices and from the map data. The latter source of errors are significant in the road network of developing cities where the roads are rapidly evolving and obtaining a reliable map is hard. Even in developed cities, the granularity of maps from different providers can vary greatly. However, it seems that the number of studies on the performances of map matching under such different level of

map granularity is relatively small.

In contrast, many methods have been proposed to cope with the sparsity and noises in the GPS data. Quite recently, a method based on the Hidden Markov Model (HMM) is shown to be effective in recovering the whole sequence of roads [8] given a sequence of GPS measurements. There is also an ongoing work of turning the algorithm into an *online* one, namely, to decide the sequence of roads incrementally as new measurements are obtained [5]. However, those methods suffer from the complexity in modelling the map matching in the HMM framework.

Our Contribution: The contribution of this paper is twofold. First, we propose a new map matching algorithm which is based on the ideal HMM. The prior work on map matching based on an HMM [8] assumes the state transition probabilities that do not exactly fit into the Viterbi algorithm, and had a drawback in finding the most likely path (as described in more details in the following section). We simplify the state transition probabilities to fit into the ideal HMM framework and to avoid the drawback. The simplification leads us to use points sampled over the road map data as hidden states in the framework. Second, we test the HMM-based methods under different levels of sampling rates of points and lines used for representing the road network and see how its performances vary. This gives an intelligent guess on the sufficient amount of road network information for extracting mobility patterns amid sparse GPS sensor data.

2. Map Matching Problem

More formally, the input and output of the map matching problem considered in this paper is as follow.

Input: A sequence of T GPS observation points, $Z = (Z_t \mid t = 1, 2, \dots, T)$, where each Z_t is composed of latitude and longitude of the moving object at time t , and a directed road network $G = (V, A)$, such that $V = (\text{cp}_i \mid i = 1, 2, \dots, N)$ is a collection of nodes that correspond to crosspoints (or, CPs) on the road network, and $A = (\text{r}_j \mid j = 1, 2, \dots, M)$ is a collection of

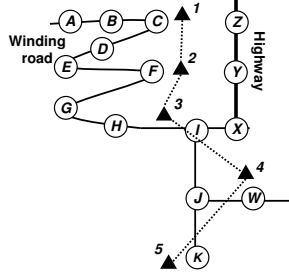


Figure 1: An example of map matching on a road network

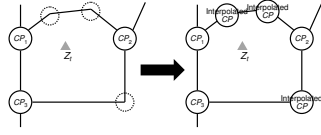


Figure 2: Representing road network data by CPs and interpolated CPs

arcs (directed edges) that correspond to road segments on the road network.

Output: A connected sequence of arcs $R = (r_i \mid i = 1, 2, \dots, T')$, where $r_i \in A$, that denotes the roads that are matched to the observed points.

A map matching instance is shown in Fig. 1, where the observed sequence of points is (1, 2, 3, 4, 5), and the road network consists of fifteen CPs, $(A, B, \dots, K, W, X, Y, Z)$, and two-way road segments that connect the CPs. In this case, there are several possible sequences of roads that can give the observed points: (1, 2, 3) can be matched to the winding road (A, B, \dots, G, H) , or the highway (Z, Y, X) , and (3, 4, 5) to the straight road (I, J, K) , or the road with u-turn (I, J, W, J, K) . The task considered in this paper is to find the most likely sequence of CPs on the roads that were traversed by the moving objects among such possible sequences.

The accuracy of the map matching also depends on the granularity of road segments in the road network. The road segments are represented by interpolated CPs as illustrated in Fig. 2, where, for example, the road between CP_1 and CP_2 is represented by interpolated CPs that are useful for representing road curves. For simplicity, we call all points as crosspoints (or, CPs) although the interpolated ones are not intersections of multiple roads. Many map data, such as the OpenStreetMap [9], represent roads as sequences of CPs. We should also note that existing methods retrieve candidate road segments by geometric projection of the observed GPS points onto the road segments between CPs.

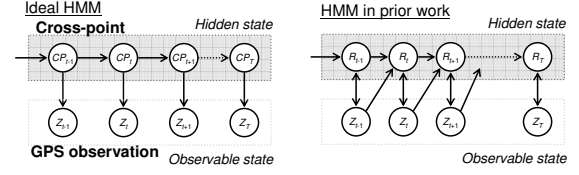


Figure 3: Comparison of HMMs

3. HMM for Map Matching

We give a brief explanation of HMM in the map matching context. In an HMM for map matching, the moving object is modeled to move according to a Markov process between CPs on the road segments. These road segments are not directly visible, and are considered as hidden states, but the output of the hidden state, namely the GPS coordinates, are. The distribution of the observed GPS coordinates depends only on the hidden state. Fig. 3 shows a diagram of an ideal HMM in the map matching, where the upper part shows the transition of hidden states (CPs) that incurs the sequence of observed outputs (GPS points).

The following probabilities are the basic components of the HMM.

Emission Probabilities: These probabilities give a conditional probability of obtaining the GPS point given the position of the moving object on the nodes of the road segment. The emission probability is denoted as $\Pr(Z_t \mid CP_t = cp_i)$, which is the probability of observing Z_t given that the (hidden) position of moving object on the road network at time t is cp_i . Following on previous work [8], the emission probability is modeled to follow a Gaussian noise centered at cp_i , and thus

$$\Pr(Z_t \mid CP_t = cp_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\|Z_t - cp_i\|_{gc}^2}{2\sigma^2}\right),$$

where $\|Z_t - cp_i\|_{gc}$ denotes the great circle distance on the surface of the earth between the true location of the moving object and the observed point. The parameter σ is the standard deviation of the GPS measurement.

State Transition Probabilities: These probabilities give a conditional probability of the object to move from a crosspoint to another crosspoint on the road network. The probabilities correspond to the transition of hidden states of the ideal HMM shown on the left of Fig. 3. We propose to set the probabilities to be proportional to the distance between the CPs on the road network, as follow.

$$\Pr(CP_{t+1} = cp_j \mid CP_t = cp_i) \propto \exp(-\beta d_{ij}),$$

where β is a paramater to control the effect of the shortest distance d_{ij} between cp_i and cp_j . The distance d_{ij}

can take into account the travel time, cost and other factors beyond the driving distance.

Initial State Probabilities: These probabilities give the likelihood of the initial CP of the moving object. In this paper, the probabilities are approximated from the emission probabilities with regard to the first GPS observation, i.e., $\Pr(Z_1 \mid \text{CP}_1 = \text{cp}_i)$.

Comparison with the previous work: The state transition probabilities in [8] are defined as

$$\Pr(R_{t+1} \mid R_t, Z_{t+1}, Z_t) \propto \exp\left(-\beta \left| \|Z_{t+1} - Z_t\|_{\text{gc}} - d_{R_t, R_{t+1}} \right| \right), \quad (1)$$

where R_t denotes a random variable for a geometric projection point on a road segment obtained by projecting Z_t , and $d_{R_t, R_{t+1}}$ is the driving distance between R_t and R_{t+1} . These state transition probabilities depend on the current and previous observed states (see the right figure of Fig. 3) and as a result do not directly give those that are needed in the Viterbi algorithm (see the left figure of Fig. 3). This is inevitable because they are designed to favor transitions whose driving distance is about the same as the great circle distance between the observed GPS points.

This creates two issues. First, to apply the Viterbi algorithm, one would, for example, need to use Eq. (1) as an approximation of $\Pr(R_{t+1} \mid R_t)$. Then, the path that is found with this approximate Viterbi algorithm does not necessarily maximize the likelihood under the state transition probabilities given in Eq. (1). Second, Eq. (1) can be seen as taking a geometric projection point R_t on the road segment as part of hidden states (beside the road segment it lies on). Clearly, such points are not discrete, and there could be various ways to obtain such a point depending on the projection rule. Thus, finding an optimal solution in the discrete HMM is difficult.

To overcome these issues, we model hidden states as (discrete) sampled points on the road segments and obtain a simple HMM that can be used to find a guaranteed optimal path. Despite its simplicity, we confirm by experiments that its performance is comparable to [8]. Moreover, from the experiments we find that the granularity of sampled points can be adjusted to obtain better matching results with more (space- and time-) efficient computation.

4. Algorithm

Given probabilities in the previous section, the most likely sequence of CPs can be computed from the ideal HMM using the Viterbi algorithm, which is a popular method in speech and text recognition tasks. The details are similar to [8], and therefore they are omitted due to the page limitation. The basic flow of the algorithm is as follow.

For each GPS point, we first find all CPs (including the interpolated ones) nearby by range query, and compute emission probabilities for each pair of the point and the CP nearby. This can be done efficiently by storing all CPs on the road network with the *kd tree* or nearest neighbor data structures. Notice that unlike other previous work, we do not project the point onto roads to retrieve CPs. This is because computing such projection is time consuming and it unnaturally assumes that emission probabilities occur only orthogonal to the road segments. We then apply the Viterbi algorithm on the GPS points and CPs. Clearly, the road segments obtained depend on the granularity of CPs on the underlying road network. This is the motivation behind our second contribution to test the method under different granularity level of road network data.

5. Experiment

We used the Microsoft dataset (MS) for testing our map-matching algorithm. The MS dataset was used in [8] and is publicly available. It consists of a 80-km trajectory of 7531 GPS points sampled at every second on the Seattle road network that consists of 418k CPs and 857k road segments. To simulate the sparseness of GPS points, we sampled them by 90, 120, 180, 240, 300, 360, 420, 480, 540, and 600 seconds ([7] pointed out that there exist huge amount of such low-sampling rate GPS trajectories, and for higher rate we believe that simpler algorithms that match points to nearest roads might be more competitive).

Because the true paths are available, we evaluated the matching results by *the route mismatched fraction* as proposed in [8]. In brief, it is derived from the ratio of the sum of the incorrect routes added to and subtracted from the correct ones against the sum of the correct ones. The HMM parameters were $\sigma = 1.0$, and $\beta = 100.0$, and for each GPS point 3 nearest CPs were retrieved for computing the emission probabilities.

The quality of the matching results is shown in Fig. 4. The figures of the route mismatch fraction of the HMM in [8] (denoted by [NK09]) were a bit higher than reported because we did not perform preprocessing steps. The HMM parameters for [NK09] were set for its optimality. From the figure, we could observe that our method (denoted by *Simplified HMM*) is comparable to [NK09] despite its simplicity.

We next simulated the sparseness of the road network data by removing 10%, 20%, 30%, 40%, 50%, 60%, 70%, and 80% of the interpolated CPs from the Seattle road network, and obtained roadnetworks with, respectively, 386k, 355k, 322k, 290k, 258k, 226k, 194k, and 162k of CPs. The average length of road segments in the resulting maps is, respectively (in meters),

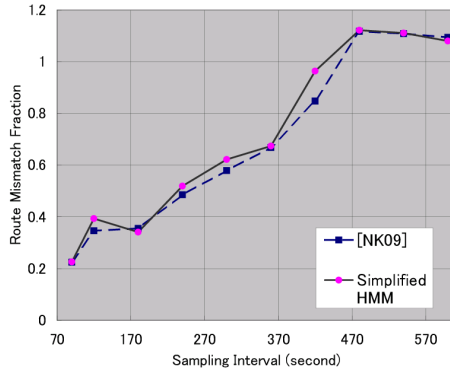


Figure 4: Map matching comparison

55, 61, 67, 72, 80, 87, 94, and 102. Performing map matching on GPS trajectories with sampling rate 90 seconds, we obtained that the map matching qualities stay roughly the same: only $\approx 9\%$ worse when the road network was sampled for every 87 meters (with 226k CPs) instead of for every 55 meters (with 418kCPs) in the full map (the average interval of GPS points in the trajectory was 863 meters). At 80-meter sampling rate, we even obtained better matching results. Interestingly, we also observed similar phenomenon on the map matching results of [NK09]. The variation might come from performing approximated Viterbi iteration only on a restricted number of CPs around the GPS points instead of the whole CPs. By having less CPs for each road segment, we could sometimes identify ground truth roads more efficiently. However, when the interval of CPs is larger than 94 meters, the quality degrades sharply.

6. Related Work

An in-depth review of map matching algorithms and their performances can be found in [10]. The offline map matching task in this paper is in the similar setting as in [12] that exploits the Integer Programming (IP). Although the IP gives an optimal matching, its computing time is prohibitive on large map and long trajectories. A method in [8] that inspires this paper is more practical because the Viterbi algorithm can be used to speed up the computation. The shortest path distance for measuring similarity of road segments was used in [13, 3]. The use of geometric approaches is also popular: [1] used Fréchet distances, and [4] used the path shapes for measuring similarity between trajectories and road curves. The geometric approaches can give better results since the matching is performed on the whole sequence, but it is rather complex. For this reason, quite recently an approximate approach has been proposed [2]. When there are sufficient number of measurement data, simple approaches that match point

to nearest road are often satisfactory, even in the online setting [6, 11].

7. Conclusion

We have presented a simple HMM-based map matching method whose performance was quite well on low-rate GPS trajectories. We have also tested the method on real-world data, and showed that it is comparable to the more-complex method. We also showed how its performance was still maintained even though the underlying road network was reduced by up to 46%. We believe this is the first work that tries to quantify the amount of road data that is sufficient for map matching under low-rate GPS trajectories. Identifying the best sampling rate of the road network for map matching is an interesting future work since the simplification can produce less accurate results.

References

- [1] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk. On map-matching vehicle tracking data. In *VLDB*, pages 853–864, 2005.
- [2] D. Chen, A. Driemel, L. J. Guibas, A. Nguyen, and C. Wenk. Approximate map matching with respect to the fréchet distance. In *ALENEX*, pages 75–83, 2011.
- [3] J. Eisner, S. Funke, A. Herbst, A. Spillner, and S. Storandt. Algorithms for matching and predicting trajectories. In *ALENEX*, pages 84–95, 2011.
- [4] S. Funke and S. Storandt. Path shapes: an alternative method for map matching and fully autonomous self-localization. In *ACM GIS*, pages 319–328, 2011.
- [5] C. Goh, J. Dauwels, N. Mitrovic, M. T. Asif, A. Oran, and P. Jaillet. Online map-matching based on hidden markov model for real-time traffic sensing applications. In *Working paper*, 2012.
- [6] J. S. Greenfeld. Matching gps observations to locations on a digital map. In *Proc. of the 81st TRB*, 2002.
- [7] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang. Map-matching for low-sampling-rate gps trajectories. In *ACM GIS*, pages 352–361, 2009.
- [8] P. Newson and J. Krumm. Hidden markov map matching through noise and sparseness. In *ACM GIS*, pages 336–343, 2009.
- [9] OpenStreetMap. www.openstreetmap.org.
- [10] M. A. Qudus, W. Y. Ochieng, and R. B. Noland. Current map-matching algorithms for transport applications: State-of-the art and future research directions. In *Transport. Res. Part C: Emerging Tech.*, volume 15, pages 312–328, 2007.
- [11] C. E. White, D. Bernstein, and A. L. Kornhauser. Some map matching algorithms for personal navigation assistants. In *Transport. Res. Part C: Emerging Tech.*, volume 8, 1.6, pages 91–108, 2000.
- [12] H. Yanagisawa. An offline map matching via integer programming. In *ICPR*, pages 4206–4209, 2010.
- [13] H. Yin and O. Wolfson. A weight-based map matching method in moving objects databases. In *SSDBM*, pages 437–438, 2004.