Route graph: joint map-matching by maximizing posterior probability

Hiroya Inakoshi[1,a)] Junichi Shigezumi[1,b)] Tatsuya Asai[1,c)] Takuya Kida[2,d)]
Hiroki Arimura[2,e)]

Abstract: We propose a joint map-matching for estimating unobservable paths from GPS traces. Our method is the first to maximize the posterior probability of stochastic generative model, in which traces are emitted as vehicles drive the roads. We employed the EM algorithm to find the parameters of the generative model, as well as to evaluate the expectations of the latent variable, which is indeed the estimated unobservable path. The EM algorithm is reduced to the exploratory search of the *route graph*, which is the geometric graph that is most likely emitting the traces and corresponds to the parameters of the model. Due to this stochastic formulation, our method works well with the presence of sampling noises in the traces. We report that the residual degradation of the estimated paths was no more than 7.0% even when they are sampled at a rate as low as 40%.

Keywords: GPS, map-matching, trajectory analysis

1. Introduction

The ability to obtain spatio-temporal information is now commonplace as vehicles and smartphones are equipped with GPS devices. On the other hand, governments, dedicated private companies, and social communities have been providing and maintaining digital road maps (DRM). Among the various information services enabled by these DRMs, the analyses of the flows of cars and people has enjoyed the most commercial success. For instance, analyzing traffic demands provides feedback for urban traffic design and the identification of typical routes improves the efficiency of distribution services [3], [8], [10], [15], [16], [20], [25].

GPS observations are collected from individual cars independently and asynchronously. Further more, they also contain observation noises, especially in urban areas with tall and large buildings. Before analyzing such irregular and unreliable GPS observations, a map-matching technique is commonly used to attach the observed trajectories on to a DRM. The authors of [4] surveyed the range of map-matching techniques, and those in [23] discussed recent developments and remaining problems.

Earlier proposals for on-line map-matching algorithms attach each observation to one of the neighboring road segments while considering the local connectivity of the segments [7], [11], [26]. Then, off-line map-matching algorithms were proposed, which consider the topological distances between trajectories and paths on a DRM [1], [6], [9], [12], [18], [19], [24], [27]. Probabilistic map-matching algorithms have also been proposed for estimating the road links from which observations are made [2], [21], [22]. Due to the limited network bandwidth or the constraint on power consumption, map-matching low-sample trajectories has attracted recent interest. One advanced algorithm utilizes observations from other trajectories to map a trajectory onto a DRM [13], [14]. Another maps trajectories to the segments embedded in a DRM all at once by formalizing map-matching as an optimization problem [17].

Most of these preceding approaches, however, mainly focused on assigning trajectories that seem natural on a DRM. By contrast, less attention has been paid to estimating unobservable paths, which are unaccessible in practical situations. In addition, we must pay more attention to identifying major streams in the trajectories to provide useful insight for realizing applications such as demand analysis and urban design, as mentioned above. In this paper, we propose a joint map-matching method, which is formulated to maximize the posterior probability of a stochastic generative model. This model represents a process in which GPS devices on vehicles generate observations as they drive along the paths, which are actually unobservable. Using this stochastic model whose latent random variable represents an occurrence of a drive on a path, our method is able to directly estimate the unobservable paths from the ob-

[1] Artificial Intelligence Laboratory, Fujitsu Laboratories Ltd.
[2] Graduate School of Information Science and Technology, Hokkaido University.
a) inakoshi.hiroya@jp.fujitsu.com
b) j.shigezumi@jp.fujitsu.com
c) asai.tatsuya@jp.fujitsu.com
d) kida@ist.hokudai.ac.jp
e) arim@ist.hokudai.ac.jp

served trajectories. Our contributions are as follows: first, we present the process that generates GPS observations and formulate it as a stochastic generative model whose latent random variable represents the occurrence of a drive on a path, and whose observed random variable represents the distance between the path and the trajectory. Second, we formulate an EM algorithm that maximizes the posterior probability of the generative model. Then, we show that the log-likelihood of the posterior probability should be reduced to an object function consisting of the residual of the trajectories from their maximizing paths and the description length of the DRM. Finally, we present our algorithm, which iteratively explores the subgraphs likely to emit the observations. The experimental results show that the residual degradation was within 7.0% even if we map-match trajectories sparsified at a rate of 40%.

The remainder of this paper is organized as follows. After providing a brief review of related works in section 2, we propose the new map-matching problem and describe its key features along with a few preliminaries in section 3. Next, we present our experimental results in section 4. Finally, we conclude the paper and suggest future work in section 5.

2. Related works

On-line or local map-matching methods attach a newly observed GPS point to one of the neighboring links in the DRM. These methods use the coordinates, directions, and speeds localized to the current point to take into account the connectivities of these links [7], [11], [26]. By contrast, off-line or global map-matching methods consider the distance between a trajectory and a path in a DRM in the topological sense, from its origin to destination [6]. Alt et al. proposed a map-matching algorithm that utilizes Fréchet distance [1]. Algorithms with the relaxed Fréchet distance have also been proposed [6], [9]. All these approaches simply map individual trajectories to the nearest paths in accordance with their own policy or distance function.

Due to the constraints on power consumption and transmission cost, trajectories are very sparse, and the above approaches do not always work well with low-sampled trajectories. To tackle these problems, probabilistic methods estimate the link from which the observation is made [19], [21], [27]. A joint map-matching method exploits the ensemble nature buried in the trajectories. It iteratively estimates the order of observations from different trajectories and the most likely segments from which they come [14]. A joint map-matching enumerates fixed-sized segments from a DRM and, using them as variables of the optimization problem, it discovers the paths to which the trajectories are assigned such that they seem to be as natural as paths [17]. This method, however, requires hyper parameters to balance the features such as the distance to the segments, stitching of segments, and regularity of the solution.

Many applications have been proposed for the analysis and prediction of traces. Some learn the repeated patterns of a car owner's history, e.g., commuting routes, the dropping off and picking up of family members, and visiting relatives or friends [10]. Turn prediction is another typical application for predicting which directions a car will take at an intersection, based on the route taken up to this point by learning others' traces [16]. These applications, however, are developed to predict particular purposes.

3. Proposed method

3.1 Preliminaries

Let $i, j, k, n, p, q, N, K \in \mathbb{N}$ and we call $G = (V, E)$ a geometric graph, or simply a graph, where $V = \{(lon, lat) \mid lon, lat \in \mathbb{R}\}$, $E = \{(u, v) \mid u, v \in V \text{ and } u \neq v\}$. Note that a geometric graph and its edges are interchangeable when we identify a graph with the edges E and the vertices given as $V = \bigcup_{(u,v)\in E}\{u, v\}$. Hereafter, we refer to this graph as an edge-induced graph $G(E)$. Given two graphs as $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, we denote $G_1 \setminus G_2 = G(E_1 \setminus E_2)$ and $G_1 \setminus E_2 = G(E_1 \setminus E_2)$. In addition, we call the element in $\mathbb{P} = \{u^{(i)} \in \mathbb{R}^2 \mid i = 0, \cdots, p\}$ a polyline of length p, or a trajectory, and the element in $\mathbb{P}_G = \{v^{(j)} \in V \mid i = 0, \cdots, q\}$ a polyline of length q on the graph G, or a path. Obviously, we have $\mathbb{P}_G \subset \mathbb{P}$.

Definition 1. *(Distance function): Let $d : \mathbb{P} \times \mathbb{P} \to \mathbb{R}$ be a distance function between polylines, where the following inequality and equalities hold for all $\alpha, \beta \in \mathbb{P}$:*

$$d(\alpha, \beta) \geq 0, \quad d(\alpha, \beta) = 0, \quad d(\alpha, \beta) = d(\beta, \alpha).$$

Definition 2. *(Single-track map-matching) Let $G = (V, E)$ be a geometric graph and $d(\alpha, \beta)$ be a distance function between polylines. Given a trajectory $\alpha \in \mathbb{P}$, a single-track map-matching algorithm, or simply a map-matching algorithm, $\mathcal{M}_G : \mathbb{P} \to \mathbb{P}_G \times \mathbb{R}$ finds its minimizing path and its minimum distance $\hat{\beta}_G(\alpha) = \text{argmin}_{\beta\in\mathbb{P}_G} d(\alpha, \beta)$ and $\hat{d}_G(\alpha) = \min_{\beta\in\mathbb{P}_G} d(\alpha, \beta)$, respectively.*

3.2 Stochastic generative model

Let $b(s) \in \mathbb{R}^2, 0 \leq s \leq q$ be a route, which is also described as a path $\beta \in \mathbb{P}_G$, such that:

$$b(s) = \begin{cases} v^{(j)} & \text{if } s = j, \\ (\lceil s \rceil - s)v^{(\lfloor s \rfloor)} + (s - \lfloor s \rfloor)v^{(\lceil s \rceil)} & \text{otherwise.} \end{cases} \tag{1}$$

Note that, without losing generality, we attached the origin and destination of the route to the first and last vertices of the path, respectively.

The observations in a trajectory $\alpha \in \mathbb{P}$ are emitted on the route $b(s)$ at $s \in \{s^{(i)} \mid i = 0, \cdots, p\}$ such that $s(i) < s(j)$ for all $0 \leq i < j \leq p$. Additionally, assuming the first and last observations are made from the origin and destination of the route, respectively, we have $s(0) = 0$ and $s(p) = q$. Each observation has its observation noise $\epsilon_i \in \mathbb{R}^2$ and thus we have:

$$u^{(i)} = b(s^{(i)}) + \epsilon_i. \tag{2}$$

Also note that a trajectory has sampling noises that are induced by interpolating the finite number of observations comprising the trajectory.

In summary, the routes are unobservable, the trajectories are observed and are emitted from one of the routes. We introduce the stochastic generative model with the observed and latent random variable $X, Z \in \mathbb{R}^K$, as follows:

Latent variable Z is a 1-of-K random variable whose realization is $z = (z_k)_{k=1}^K$, where $z_k \in \{0, 1\}$. There is a k^* such that $z_k = 1$ if $k = k^*$ and $z_k = 0$ otherwise, which represents that a drive on the k^*-th route out of K possible routes occurred. Its occurrence follows the prior probability distribution of $P(z) = \prod_{k=1}^K {\pi_k}^{z_k}$ such that $\sum_{k=1}^K \pi_k = 1$.

Observed variable X is a random variable whose realization is $x = (x_k)_{k=1}^K$, which represents the distance between trajectory and path. This distance follows the probability distribution of $P(x|z) = \prod_{k=1}^K \mathcal{N}(x_k|0, \sigma^2)^{z_k}$. For simplicity, the mean and deviation are independent of k and known to be 0 and σ^2, respectively. σ differs in accordance with the extent of the sampling noises.

3.3 Maximizing posterior probability

Given a collection of traces $T = \{\alpha_n \mid n = 1, \cdots, N\}$ and if let x_n and z_n be the independent realizations of the random variables X and Z, respectively, we have the concrete E-step and M-step for the joint map-matching by applying the above probability distributions to the generic EM algorithm in a similar manner as that for a Gaussian mixture model [5]. With an initial π^{old}, it iterates the following E-step and M-step by replacing π^{old} with π^{new} until either Q' or π converges:

E-step evaluates the responsibility $\gamma(z_{nk})$ with the parameter π^{old}, and

M-step finds π^{new} that maximizes the log-likelihood of posterior probability $Q'(\pi, \pi^{\text{old}})$.

The responsibility and the log-likelihood are respectively defined as follows:

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x_{nk}|0, \sigma^2)}{\sum_{k'=1}^K \pi_{k'} \mathcal{N}(x_{nk}|0, \sigma^2)}, \quad (3)$$

$$Q'(\pi, \pi^{\text{old}}) = -\sum_{n=1}^N \sum_{k=1}^K \gamma(z_{nk}) \frac{x_{nk}}{2\sigma^2} + \sum_{k=1}^K z_k \ln \pi_k. \quad (4)$$

Although there are too many paths on the graph, it is practically sufficient to consider the paths that have shorter distances from each trajectory. This is feasible if we employ an algorithm [24] that can enumerate all the paths whose distances from the trajectory are within a certain threshold, such as σ. In extreme, considering just the minimizing path $\beta_{k^*} = \hat{\beta}_G(\alpha)$, we have $\gamma(z_{nk}) = 1$ if $k = k^*$ and $\gamma(z_{nk}) = 0$ otherwise. Assuming that the prior distribution is uniform, namely $\pi_k = 1/K$ for all k, the second term of Eq. (4) is straightforward and equal to $-K$. If we accept that K is roughly proportional to the description length of the geometric graph G, the joint map-matching is equivalent to the minimization problem below.

Definition 3. *(Route graph discovery) Let a hypothesis space of a graph be $\mathbb{G}$, a single-track map-matching be $\mathcal{M}_G$, and a collection of trajectories be T. A graph $G \in \mathbb{G}$ most likely emits the trajectories T if it minimizes the following loss function:*

$$L(G; T) = \sum_{\alpha \in T} \hat{d}_G(\alpha) + \lambda \|G\|, \quad (5)$$

where $\|G\|$ is the description length of the graph G, such as the total length of its edges, and $\lambda > 0$ is a hyper parameter.

The first term is for the residual and the second term is for the regularization. This problem is equivalent to single-track map-matchings if λ is zero. Otherwise, some edges are left unused so that $\|G\|$ decreases even though the distance $d(\alpha, \beta)$ becomes longer for some trajectories.

3.4 Graph exploration algorithm

To minimize L(G;T), we employ an exploratory search in the graph space, and obtain a decreasing series of graphs $G^{(t-1)} \supset G^{(t)}$ for $t = 1, 2, \cdots$ such that their losses also decreases. Let us denote the output of map-matching $\mathcal{M}_{G^{(t)}}$ $\hat{\beta}_\alpha^{(t)} = \hat{\beta}_{G^{(t)}}(\alpha)$ and $\hat{d}_\alpha^{(t)} = \hat{d}_{G^{(t)}}(\alpha)$ for short.

Before presenting the important property that drives the exploration, we note that $\hat{d}_\alpha^{(t-1)} \leq \hat{d}_\alpha^{(t)}$ always holds. This is trivial because if there were a path closer to α in $G^{(t)}$, it must be closer to α than the minimizing path in $G^{(t-1)}$ and this is contradictory. We do not care how the map-matching is implemented as long as it satisfies the inequality above.

The following theorem gives the condition that ensures that decreasing series of graphs decreases their losses. Interested readers can find the proof in the full-paper.

Theorem 1. *Given a collection of trajectories T, and two graphs $G^{(t-1)}$ and $G^{(t)}$, $L(G^{(t)}; T) < L(G^{(t-1)}; T)$ holds iff the following inequality holds:*

$$\lambda \|\Delta^{(t)}\| > \sum_{\alpha \in T_{|\Delta^{(t)}}} \left\{ \hat{d}_\alpha^{(t)} - \hat{d}_\alpha^{(t-1)} \right\}, \quad (6)$$

where $\Delta^{(t)} = G^{(t-1)} \setminus G^{(t)}$ and $T_{|\Delta^{(t)}}$ denotes the collection of trajectories whose minimizing paths run through $\Delta^{(t)}$.

Alg. 1 is the pseudo-code for the route graph discovery. Given a collection of trajectories T and an initial graph $G^{(0)}$, e.g., a DRM, it finds the final graph that minimizes the loss (line 7). Note that B maintains a minimizing path and the minimum distance for each trajectory α throughout every t-th stage (line 18). The main loop (line 2–6) explores a series of subgraphs with decreasing losses as explained in **Thm. 1**. First, an edge e is selected, for instance, in increasing order of the cardinality, which is the number of the minimizing paths running through it (line 3), and a new graph $G^{(t)}$ is obtained by *re-routing* with the edge e disabled (line 4). We explain later what re-routing is. Then, the graph $G^{(t)}$ is probabilistically accepted or rejected (line 5). Finally, the main loop either continues or breaks in accordance with the history of the obtained graphs (line 6).

Algorithm 1 Building route graph

Require: trajectories T, minimizing paths B

1: **procedure** MAIN($G^{(0)}$)
2: **for all** $t = 1, 2, 3, \cdots$ **do**
3: select e from edges in $G^{(t)}$
4: $G^{(t)} \leftarrow$ APPLY(e, $G^{(t-1)}$)
5: accept $G^{(t)}$ with some probability
6: break by history $\cdots, G^{(t-1)}, G^{(t)}$
7: report $G^{(t)}$
8: **function** APPLY(e, $G^{(t-1)}$)
9: let $\Delta^{(t)}$ be subgraph only $T_{|\{e\}}$run
10: $\epsilon \leftarrow 0$
11: **for all** $\alpha \in T_{|\{e\}}$ **do**
12: $\hat{\beta}_\alpha^{(t-1)}, \hat{d}_\alpha^{(t-1)} \leftarrow B[\alpha]$
13: $\hat{\beta}_\alpha^{(t)}, \hat{d}_\alpha^{(t)} \leftarrow \mathcal{M}_{G\setminus\{e\}}(\alpha)$
14: $\Delta^{(t)} \leftarrow \Delta^{(t)} \setminus \hat{\beta}_\alpha^{(t)}$
15: $\epsilon \leftarrow \epsilon + (\hat{d}_\alpha(t) - \hat{d}_\alpha^{(t-1)})$
16: **if** $\|\Delta^{(t)}\| < \epsilon$ **then**
17: **return** $G^{(t-1)}$
18: update $B[\alpha]$ with $\hat{\beta}_\alpha^{(t)}, \hat{d}_\alpha^{(t)}$
19: **return** $G^{(t-1)} \setminus \Delta^{(t)}$

Next, we explain how to obtain the new graph $G^{(t)}$ by finding $\Delta^{(t)}$. The re-routing technique serves this purpose by map-matching with some edges of graph $G^{(t)}$ disabled (line 13). Note that trajectories not in $\Delta^{(t)}$ is irrelevant to $\Delta^{(t)}$ and that any edge $e \in \Delta^{(t)}$ satisfies the following inequality:

$$\{e\} \subseteq \Delta^{(t)} \subseteq \overline{\Delta}^{(t)}, \tag{7}$$

$$\overline{\Delta}^{(t)} = G^{(t-1)} \setminus \bigcup_{\alpha \notin T_{|\{e\}}} \hat{\beta}_\alpha^{(t-1)}, \tag{8}$$

where $T_{|\{e\}}$is the collection of trajectories whose minimizing paths run through the edge e. Thus, the following strategy works: first conservatively select an edge e from $G^{(t)}$ and optimistically initialize $\Delta^{(t)}$ with $\overline{\Delta}^{(t)}$ (line 9), as well as the cumulative differential residual ϵ with 0 (line 10). Then, as we re-route a trajectory in $T_{|\{e\}}$, $\Delta^{(t)}$ is subtracted by $\hat{\beta}_\alpha^{(t)}$, and ϵ is added by the differential residual before and after the re-routing (line 13–15). In this implementation, we employed Zeheng et al.'s algorithm [27] for re-routing.

The procedure terminates as soon as it becomes obvious that Eq. (6) will never be satisfied (line 16,17). This is safe because of **Thm. 1** and, notably, this saves much computation by skipping unnecessary map-matchings. If no early termination has occurred, the procedure returns with the new graph reduced by $\Delta^{(t)}$ as $G^{(t)}$ (line 19).

4. Experimental Results

In this section, we examine whether our algorithm is able to estimate unobservable paths from sampled trajectories using the benchmark datasets. First, we explain the experimental configurations and then present the results.

4.1 Experimental configurations

We use the benchmark datasets of GPS traces, which are collected and shared by various research groups or volunteers. The DRM should be contemporary with the GPS traces, although we utilized the Open Street Map (OSM) of 2017. **Figure 1** shows descriptions of the datasets and their corresponding DRMs. Note that they might differ from those in other reports because those datasets were preprocessed differently.

	GPS		OSM	
name	#pts	#trace	#nodes	#edges
icdm	2859950	4257	18716	35170
chicago	118360	889	46533	88942
bikely	549920	3150	262699	540017

Fig. 1 Popular GPS trace datasets

In the experiments, we sampled observations in a trajectory-wise manner with variable rates, and compared the residuals of the following:

(1) the unsampled trajectory and the DRM (lower bound),
(2) the unsampled trajectory and the route graph (proposed),
(3) the sampled trajectory and the DRM (upper bound).

The first situation gives the lower bounding residual, in that no algorithm can do better than this method, as we had accepted the three assumptions made in the full-paper. The second method is our proposed method. To evaluate how well the route graph represents the major streams in the GPS traces compared to the DRM, we evaluated the residual of the unsampled trajectory and the route graph. The third method evaluates the residual arising from the injection of sample noises. This gives the upper bounding residual in the sense that no off-the-shelf single-track map-matching does worth than this.

4.2 Experimental results

Figure 2 describes the residuals of the three methods with variable sampling rates. Note that the lower bound is constant because the first method is unaffected by the sampling rates. In the figure, we can see that the upper bound curve steeply increases as the sampling rate decreases. This is what we expected, as the lower is the sampling rate, the more the residual experiences injected sampling noises.

In contrast, the curve for the proposed method increases much more moderately. For instance, in the `icdm` and `chicago` datasets, the proposed method reduced the residual by more than 70% and 40%, respectively, for the upper bound at sampling rate of 40%. The reduction rate tends to increase when the dataset has a larger number of trajectories, which means that the proposed method leverages the residual using the other trajectories. Indeed, we can see that the `icdm` was able to decrease the sampling rate to 40% while its residuals remained nearly constant at 7.0%.

The residual is slightly larger than the upper bound at a sampling rate of 100%, this is because our algorithm accepts a slight increase in the residuals to reduce the route graph. The behavior of the residual as well as the empirical loss is well understood in the regularization technique. The results may accordingly indicate that our algorithm may further exploit the observation and sampling noises.

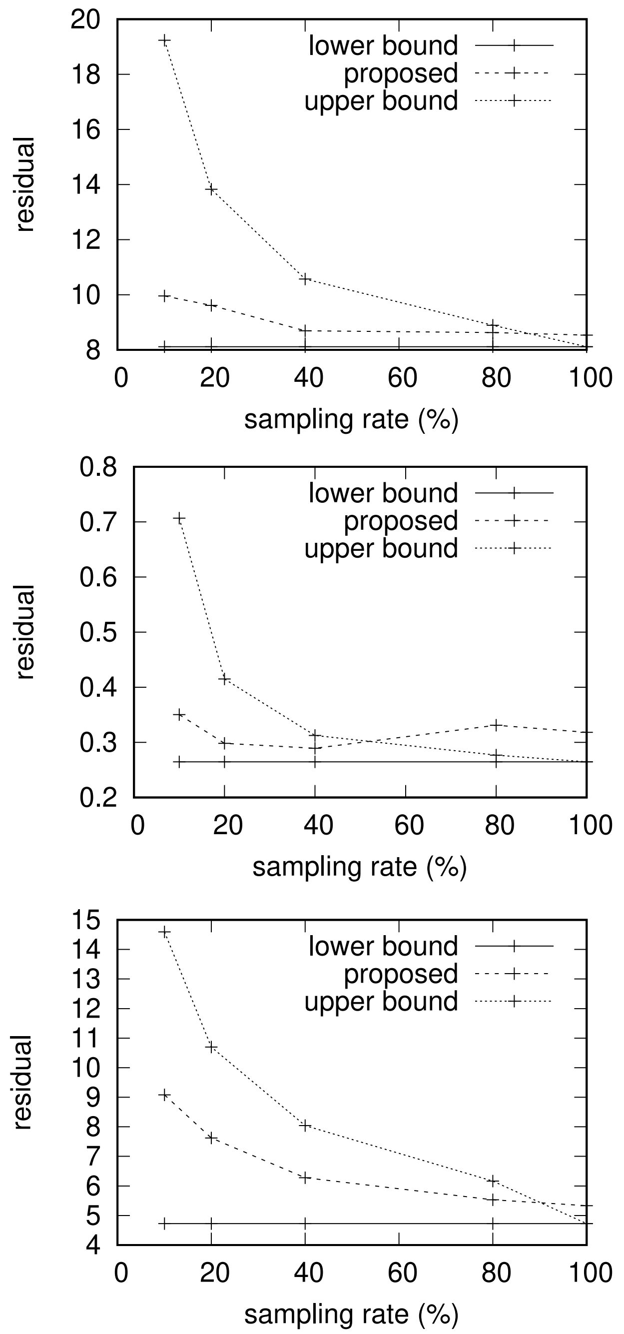

Fig. 2 Residuals of icdm (top), chicago (middle), and bikely (bottom).

5. Conclusion

In this paper, we proposed a joint map-matching method based on the generative model for estimating unobservable paths by maximizing the posterior probability. Maximization is achieved by the EM algorithm whose object function consists of residual and regularization terms. We presented an iterative algorithm for exploring the route graph, which avoids as many map-matchings as possible by taking advantage of the proven property holding of the residual and the regularization terms. The experimental results showed that the residual degradations from the lower bound were no more than 7.0% when the sampling rate was reduced to 40%. This means that this algorithm reduces the volume of sampling noises and identifies the major streams in the trajectories.

In the future, we plan to continue this work in three directions: first, by realizing performance enhancements by further reducing the costly map-matching. One idea is to localize the re-routing to the disabled links without performing map-matching from the origin to destination of the trajectories. The other idea is to extend our algorithm to the on-line that updates the route graph as trajectories arrive in sequence. The second direction is to develop more sophisticated formulations of the EM algorithm without considering the extreme case, or applying another generative model such as the Hidden Markov Model. The third and final direction is to apply our method to demand analysis, urban design, and other applications.

References

[1] Alt, H. and Godau, M.: Computing the Fréchet distance between two polygonal curves, *International Journal of Computational Geometry & Applications*, Vol. 05, No. 01n02, pp. 75–91 (online), DOI: 10.1142/S0218195995000064 (1995).

[2] Alvarez-Garcia, J., Ortega, J., Gonzalez-Abril, L. and Velasco, F.: Trip destination prediction based on past GPS log using a Hidden Markov Model, *Expert Systems with Applications*, Vol. 37, No. 12, pp. 8166–8171 (online), DOI: 10.1016/j.eswa.2010.05.070 (2010).

[3] Benkert, M., Gudmundsson, J., Hübner, F. and Wolle, T.: Reporting flock patterns, *Computational Geometry*, Vol. 41, No. 3, pp. 111–125 (online), DOI: 10.1016/j.comgeo.2007.10.003 (2008).

[4] Bernstein, D. and Kornhauser, A.: An introduction to map matching for personal navigation assistants, *The Transportation Research Board 77th Annual Meeting*, Washington, D.C (1996).

[5] Bishop, C. M. and M., C.: *Pattern recognition and machine learning*, Springer (2006).

[6] Böhm, K., Pfoser, D., Salas, R. and Wenk, C.: On map-matching vehicle tracking data, *Proceedings of the 31st international conference on Very large data bases*, ACM, pp. 853–864 (2005).

[7] Chawathe, S. S.: Segment-Based Map Matching, *2007 IEEE Intelligent Vehicles Symposium*, IEEE, pp. 1190–1197 (online), DOI: 10.1109/IVS.2007.4290280 (2007).

[8] Chen, L., Lv, M. and Chen, G.: A system for destination and future route prediction based on trajectory mining, *Pervasive and Mobile Computing*, Vol. 6, No. 6, pp. 657–676 (online), DOI: 10.1016/J.PMCJ.2010.08.004 (2010).

[9] Eiter, T. and Mannila, H.: Computing discrete Fréchet distance, *Notes*, Vol. 94, p. 64 (online), DOI: 10.1.1.90.937 (1994).

[10] Froehlich, J. and Krumm, J.: Route Prediction from Trip Observations, (online), DOI: 10.4271/2008-01-0201 (2008).

[11] Greenfeld, J.: Matching GPS observations to locations on a digital map, *Transportation Research Board 81st Annual Meeting*, Washington D.C. (2002).

[12] Hsueh, Y.-L. and Chen, H.-C.: Map matching for low-sampling-rate GPS trajectories by exploring real-time moving directions, *Information Sciences*, Vol. 433-434, pp. 55–69 (online), DOI: 10.1016/J.INS.2017.12.031 (2018).

[13] Javanmard, A., Haridasan, M. and Zhang, L.: Multi-track Map Matching, (online), available from ⟨http://arxiv.org/abs/1209.2759⟩ (2012).

[14] Javanmard, A., Haridasan, M. and Zhang, L.: Multi-track map matching, *Proceedings of the 20th International Conference on Advances in Geographic Information Systems - SIGSPATIAL '12*, New York, New York, USA, ACM Press, p. 394 (online), DOI: 10.1145/2424321.2424373 (2012).

[15] Krumm, J., Gruen, R. and Delling, D.: From destination prediction to route prediction, *Journal of Location Based Services*, Vol. 7, No. 2, pp. 98–120 (online), DOI: 10.1080/17489725.2013.788228 (2013).

[16] Krumm, J. and John: Where will they turn: predicting turn proportions at intersections, *Personal and Ubiquitous Computing*, Vol. 14, No. 7, pp. 591–599 (online), DOI: 10.1007/s00779-009-0248-1 (2010).
[17] Li, Y., Huang, Q., Kerber, M., Zhang, L. and Guibas, L.: Large-scale joint map matching of GPS traces, *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems - SIGSPATIAL'13*, New York, New York, USA, ACM Press, pp. 214–223 (online), DOI: 10.1145/2525314.2525333 (2013).
[18] Liu, X., Biagioni, J., Eriksson, J., Wang, Y., Forman, G. and Zhu, Y.: Mining large-scale, sparse GPS traces for map inference, *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '12*, New York, New York, USA, ACM Press, p. 669 (online), DOI: 10.1145/2339530.2339637 (2012).
[19] Lou, Y., Zhang, C., Zheng, Y., Xie, X., Wang, W. and Huang, Y.: Map-matching for low-sampling-rate GPS trajectories, *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems - GIS '09*, New York, New York, USA, ACM Press, p. 352 (online), DOI: 10.1145/1653771.1653820 (2009).
[20] Miklušćák, T., Gregor, M. and Janota, A.: Using Neural Networks for Route and Destination Prediction in Intelligent Transport Systems, Springer, Berlin, Heidelberg, pp. 380–387 (online), DOI: 10.1007/978-3-642-34050-5-43 (2012).
[21] Newson, P. and Krumm, J.: Hidden Markov map matching through noise and sparseness, *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems - GIS '09*, New York, New York, USA, ACM Press, p. 336 (online), DOI: 10.1145/1653771.1653818 (2009).
[22] Pink, O. and Hummel, B.: A statistical approach to map matching using road network geometry, topology and vehicular motion constraints, *2008 11th International IEEE Conference on Intelligent Transportation Systems*, IEEE, pp. 862–867 (online), DOI: 10.1109/ITSC.2008.4732697 (2008).
[23] Quddus, M. A., Ochieng, W. Y., Zhao, L. and Noland, R. B.: A general map matching algorithm for transport telematics applications, *GPS Solutions*, Vol. 7, No. 3, pp. 157–167 (online), DOI: 10.1007/s10291-003-0069-z (2003).
[24] Shigezumi, J., Asai, T., Morikawa, H. and Inakoshi, H.: A Fast Algorithm for Matching Planar Maps with Minimum Fréchet Distances, *Proceedings of the 4th International ACM SIGSPATIAL Workshop on Analytics for Big Geospatial Data - BigSpatial'15*, New York, New York, USA, ACM Press, pp. 25–34 (online), DOI: 10.1145/2835185.2835186 (2015).
[25] Tiwari, V. S., Arya, A. and Chaturvedi, S.: Route prediction using trip observations and map matching, *2013 3rd IEEE International Advance Computing Conference (IACC)*, IEEE, pp. 583–587 (online), DOI: 10.1109/IAdCC.2013.6514292 (2013).
[26] Wenk, C., Salas, R. and Pfoser, D.: Addressing the Need for Map-Matching Speed: Localizing Globalb Curve-Matching Algorithms, *18th International Conference on Scientific and Statistical Database Management (SSDBM'06)*, Vienna, Austria, IEEE, pp. 379–388 (online), DOI: 10.1109/SSDBM.2006.11 (2006).
[27] Zheng, Y., Xie, X., Zhang, C., Lou, Y., Wang, W. and Huang, Y.: Map-Matching for Low-Sampling-Rate GPS Trajectories (2009).