

不完全な道路ネットワークを用いたマップマッチングおよび道路ネットワークの補間手法の提案 Map Matching and Interpolation for Incomplete Road Networks

余 家豪[†] 佐々木 勇和[‡] 石川 佳治[†]
Yu Jiahao[†] Yuya Sasaki[†] Yoshiharu Ishikawa[‡]

1. はじめに

近年、カーナビゲーションシステムやスマートフォン等、GPS を搭載しているデバイスが一般的となっている。GPS での位置測位では、誤差を含んでいることが多く、正確に位置を測定することは難しい。誤差を修正するための技術として、マップマッチングがある [2, 3]。マップマッチングは、基本的に車載 GPS を考慮し、自動車は道路上のみを移動するという前提に GPS の位置情報を道路上に修正するという技術である。多くのマップマッチングアルゴリズムでは、完全な道路ネットワーク（つまり、全ての道路が欠損していない道路ネットワーク）を想定している。しかし、不完全な道路ネットワークも世の中には数多く存在している。例えば、OpenStreetMap のような手動にて更新される道路図や、発展途上国の道路図、GPS データと道路図の取得日時に差がある場合などが考えられる。完全な道路ネットワークを想定しているマップマッチングアルゴリズムに対して、不完全な道路ネットワークを入力とすると、大きな誤差を含んだ結果となってしまふ。図 1 は、道路ネットワーク上に人工の GPS データ（赤色の点）をプロットしたものである。図の左上の 2 箇所において、道路上を移動していない GPS データが存在する。図 2 に、完全な道路ネットワークを想定したマップマッチングアルゴリズムを適用した結果を示す。緑色の点でマップマッチングの結果を示すが道路ネットワークにない道路を通った場合、マップマッチング結果の点間の距離が非常に離れているような普通ではありえない結果が返ってくる。

一方、不完全な道路ネットワークを考慮したマップマッチングアルゴリズムも存在する [1]。この手法では、道路図には存在しないが GPS データが通過する軌跡を **off-road** として取り入れ、ある程度の距離が道路ネットワークから離れた場合に道路上以外を移動していると判断し、GPS データをそのまま出力する。3 は、off-road を考慮したマップマッチング結果を示す。図 2 の結果とは異なり、道路上を移動していない判断された点においては、そのまま出力されることにより、自然な移動軌跡が出力されていることがわかる。off-road を考慮した手法では、単純にひとつの GPS データの軌跡をマッチングすることのみを考えており、道路ネットワークをより完全なものに近づけていくことは考えていない。そこで、本研究では、不完全な道路ネットワークを考慮したマップマッチングを実行しながら、道路ネットワークを補間していくことを目的とする。単純に、出力された off-road を道路とする場合、2つの問題がある。まず、一つ目は、実際に道路が無いにも関わらず、GPS の誤差により off-road と判定される場合である。この場合、道路を補間することは不適切である。一つの軌跡だけで道路かどうかを判定するのではなく、複数の GPS データを用いることが重要である。二つ目は、複数の off-road が似たような位置に出力される場合である。近い位置に複数の道路が存在することは稀であるため、複数の off-road から一つの道路を推定する必要がある。しかし、実際に複数の道路を補間する必要がある可能性もあり、周りの道路との距離等から求める必要がある。本研究では、DBSCAN[4] を用いて、off-road の軌跡をクラスタリ

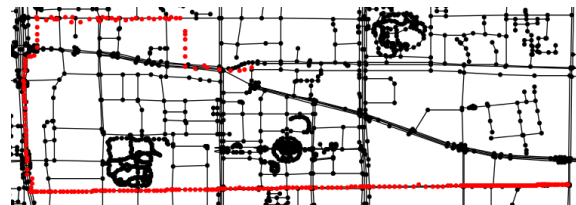


図 1: 不完全な道路ネットワーク上の GPS 軌跡

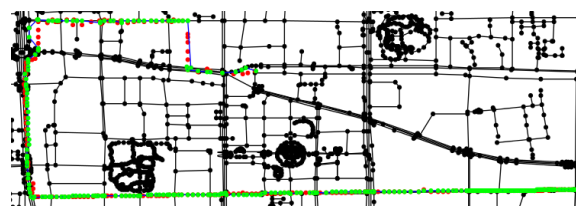


図 2: 不完全な道路ネットワークを考慮しないマップマッチングの結果

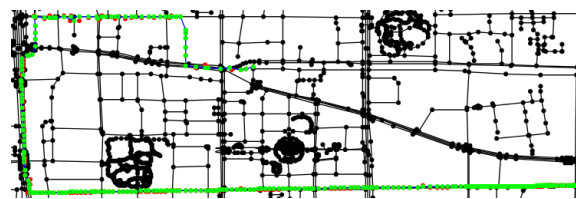


図 3: 不完全な道路ネットワークを考慮したマップマッチングの結果

ングし、クラスタリングした GPS 点を回帰分析することにより道路図を推定する。DBSCAN の際にパラメータを GPS データと道路ネットワークから求めることにより、実際に道路が存在するか、もしくは何本存在するかを推定する。

2. 事前準備

本稿では、オフラインのマップマッチングを想定する。つまり、GPS データの軌跡は始点から終点まで既に全て揃っており、新たな GPS 点が追加されることはない。GPS 点は、誤差を含む緯度・経度の情報、測定された時間情報、および機器の識別子から成っている。どの程度の誤差を含むかは自明ではないとする。GPS 点の集合を、識別子毎に時間軸順に並べることにより、軌跡を求めることが可能である。道路ネットワークは、節点と枝から構成され、節点は緯度・経度情報を保持している。そのため、GPS 点が道路ネットワーク上の節点もしくは枝からどの程度離れているかを計算することが可能である。この道路ネットワークは、不完全な道路ネットワークであり、一部が欠損していることを想定する。ただし、完全な道路ネットワークの場合でも動作し、この場合マップマッチング実行するだけとなる。

3. 関連研究

本章では、完全な道路ネットワークおよび不完全な道路ネットワークにおけるマップマッチングに関する研究をそれぞれ紹介する。

完全な道路ネットワークにおけるマップマッチングでは、Lou らの ST-Matching アルゴリズム [2] と、Newson らの Hidden Markov Model に基づく手法 [3] がある。両方の手法とも、まず、GPS 点から道路ネットワーク上の複数個の点を候補点とし

[†]名古屋大学大学院情報科学研究科
Graduate School of Information Science, Nagoya University
[‡]大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology,
Osaka University

て抽出する。各GPS点の候補点を時間軸上に並べ、それらをつなぐことにより、候補点グラフ G_{cand} を構築する。二つのアルゴリズムでは、各候補点間の枝の重みが異なる。ST-Matching アルゴリズム [2] では、空間と時間の連続性をもとに枝の重みを決定する。Hidden Markov Model に基づく手法 [3] では、確率密度などをもとに枝に確率的な重みを与える。最後に、両手法とも候補点グラフにおける、経路上の重みが最も大きくなる一本の候補経路を取り出し、それをマッチした軌跡とする。これらの手法では、候補点グラフは、道路上の点しか考慮していないため、GPS点をそのまま出力することはない。

不完全な道路ネットワークを考慮した手法として、Hidden Markov Model に基づく手法を Jan-Henrik らが拡張している [1]。[1] のアイデアは、候補点グラフに、GPS点をそのまま候補点として追加することである。候補点グラフ G_{cand} においては、枝の重みを計算する上で、三種類の要素: (1) GPS点と候補点間のユークリッド距離、(2) 道路上の候補点における道路ネットワーク上での最短経路距離、および (3) 連続する2つの候補点が生じるGPS点か道路上に修正した候補点かにより場合分けした値を考える。三つの要素の値を掛け合わせるものを枝の重みとする。候補点グラフを作成した後は、他の手法と同様に一本の候補経路を取り出し、マッチング結果とする。

4. 提案手法

本手法では、[1]の手法をもとに off-road のGPS点から道路ネットワークを補間する。そのために、まずマップマッチングした結果から off-road とされたGPS点を抜き出す。抜き出したGPS点集合をDBSCAN[4]を用いてクラスタリングすることにより、補間する道路の本数を決定する。クラスタリングされたGPS点を回帰分析することにより補間する道路を決定する。

4.1 クラスタリング

DBSCANでは、道路ネットワークの構造などは考慮せずに、基本的に距離のみから計算している。道路ネットワークやGPS点の軌跡の特徴を用いて、ヒューリスティックにクラスタリングを実現する。まず、不完全な道路ネットワークに欠けている道路は、高速道路のような大きな道路ではなく、あまり人が通らない細い道路であることが一般的である。そのため、既存の道路をまたぐような大きな道路ではないということを想定し、クラスタリングの事前準備として、道路ネットワークに囲まれている領域ごとにGPS点の軌跡に分割する。異なる道路ネットワークに囲まれたGPS点の軌跡を同じクラスタに含まれるないようにする。

次に、DBSCANを用いてクラスタリングする場合、パラメータとして最小点数 $MinPts$ と半径 ϵ を決定する必要がある。半径 ϵ は、道路ネットワークにおける道路間の距離に密接するパラメータである。そのため、与えられた道路ネットワークの道路間の距離に近い場合は、小さい値をとるべきと考えられる。このパラメータをユーザが決定することは簡単ではなく、入力された道路ネットワークから自動で求めることが要求される。そこでまず、off-road と判定されたGPS点の軌跡における、GPS間の最大距離を m_{off} 。次に、全ての道路上にマッチングされた点とそれに対応するGPS点間の距離の平均 a_{err} を求める。これは、GPSの平均誤差とみなすことができる。まず、同じ軌跡上のデータは、同じクラスタに属するべきであるため、 m_{off} が ϵ の最小の値となる。次に、誤差が大きい場合、2つの軌跡が同じ道路を通っているにも関わらず、同じクラスタに属しない可能性がある。そこで、半径 ϵ は、これらの値の最大値をとる $\max(2 \cdot a_{err}, a_{ve_{off}})$ とする。つまり、この値は少なくとも同じGPS軌跡データは同じクラスタに含まれ、さらに平均誤差がそれよりも大きい場合、二つの異なるGPS軌跡データが同じ道路上を通過していると判定するものである。

次に、最小点数 $MinPts$ を求める。自身を中心とする半径 ϵ の円内に最小点数よりも多くの点(自身も含む)が存在する点は、コア点とされ、コア点と半径 ϵ 以内に存在するものは同じクラスタに含まれる。この最小点数は、道路を補間するかどうかの閾値とみなすことができる。まず、少なくとも、同じ軌跡のGPS点は ϵ 以内に必ず存在するよう設定したため、最小点数 $MinPts$ は少なくとも3以上となる。ここで、重要な点は、自身以外の軌跡が ϵ に存在するかどうかである。そこで、新たなパラメータとして最小軌跡数 $MinTrj$ を定義する。 ϵ に最小軌跡数以上の存在する点をコア点とするよう拡張する。 $MinTrj$ は、追加する補間された道路の信頼度のようなものを示すパラメータであり、ユーザが決定するものとする。もし $MinPts$ を3とし、 $MinTrj$ を1とした場合、全ての off-road を道路として補間することを意味する。

4.2 道路ネットワークの補間

クラスタリングにより、道路を追加すべき部分のGPS点の集合を得ることができる。このGPS点から回帰分析を用いることにより、補間する道路ネットワークを決定する。回帰により求めた道路ネットワークがGPS点とどの程度離れているか、つまり誤差がどの程度であるかを求める必要がある。例えば、新たに補間する道路と補間に使われたGPS点の誤差が大きい場合、その道路の補間は間違っている可能性が高い。本研究では、道路が基本的には直線が多いと考え、最小二乗モデルによる線形回帰を適用することを考える。しかし、道路が非線形(つまりカーブを含む場合)や曲がり角含む場合、クラスタリング結果をさらに分割かついて道路を求める必要がある。これを繰り返し、最大誤差が off-road 以外のマップマッチング結果より小さく、平均的な誤差が小さいように求める必要がある。

5. まとめ

本稿では、不完全な道路ネットワークを用いたマップマッチングと道路ネットワークの補間手法を提案した。不完全な道路ネットワークにマップマッチング手法 [1] を利用して、off-road 軌跡を抜き出し、抜き出したデータをクラスタリングし、道路ネットワークを補間する。今後、まず提案手法を実データをを用いて評価する。さらに、大規模なデータを用いた場合、実行時間が大きくなることが予想される。そこで、効率的に手法をマップマッチングおよび道路ネットワークの補間を実行する手法を考案する予定である。

謝辞

本研究の一部は、科研費(15K21069, 16H01722, 26540043)および文部科学省「実社会ビッグデータ利活用のためのデータ統合・解析技術の研究開発」による。

参考文献

- [1] J.-H. Huanert and B. Budig, "An Algorithm for Map Matching Given Incomplete Road Data", In *Proc. ACM SIGSPATIAL GIS*, pp. 510–513, 2012.
- [2] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang, "Map-Matching for Low-Sampling-Rate GPS Trajectories", In *Proc. ACM SIGSPATIAL GIS*, pp. 352–361, 2009.
- [3] P. Newson and J. Krumm, "Hidden Markov Map Matching Through Noise and Sparseness", In *Proc. ACM SIGSPATIAL GIS*, pp. 336–343, 2009.
- [4] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", *Proc. SIGKDD*, pp. 226–231, 1996.