

CHANTAKU : 多様なフィルタリングとスケール・レンジ フリーな閲覧を実現するライフログ可視化システム

北沢 匠¹ 西尾 信彦²

概要: 携帯端末の普及と高機能化に伴い、そこから得られるライフログを蓄積・解析することで、端末を持つユーザの生活習慣分析や、行動予測をする研究が行われている。このような研究を行う際には、センサーデータの解析やマイニングプログラムのデバッグといった工程が発生し、多くの場合、研究者はデータをグラフとして描画する可視化システムを用いる。しかし、現状存在するライフログの可視化システムを利用した場合、指定されたデータの期間やサンプル数(以下、レンジ)と、グラフや二次元地図などの可視化結果の縮尺(以下、スケール)を考慮した可視化が行われず、データによっては可視化結果がみづらくなってしまうといった問題が存在する。本研究ではこの問題に対し、ユーザに指定されたデータをレンジや、可視化結果のスケールに合わせて自動的に間引く機能を備えたライフログ可視化ツールであるCHANTAKUを設計・実装した。また、この間引き処理はユーザによって許容できる程度が異なるため、処理に用いる閾値を変更可能にした。さらに、この閾値変更を快適に行うために、間引き処理結果をプリフェッチする機能を設け、キャッシュの有無で処理速度の比較を行い、有用性を確認した。

キーワード: ライフログ, 可視化

CHANTAKU : The lifelog visualization system for scale free analyzing

KITAZAWA TAKUMI¹ NISHIO NOBUHIKO²

Abstract: In recent years, we have been able to get various data from mobile phones. These data are supposed to be life-log. Many researchers attempt to reflect user's lifestyle using such data. In doing so, the researchers have to analyze the data. In the analyzing process, it is hard to understand the user's lifestyle refer to the life-log and mining results as numerical value and character string. Thus, the researchers often use data visualization system. However, existing life-log visualization system use only one visualizing method no matter which size or term of data. Therefore, in some data, visualized results are indistinct. In this research, we propose life-log visualization system to solve the preceding problem. The purposed system can simplify data for visualizing large size and long term data. Moreover, the user can change threshold value with simplification. We also implemented prefetching function for smooth scale change and usefulness was positively evaluated.

Keywords: life-log, visualization

1. 序論

近年の携帯端末の高機能化は目覚ましく、GPS レシーバなどの測位デバイス、Wi-Fi や Bluetooth などの無線通信デバイス、加速度センサやジャイロセンサなどのセンサデバイスが搭載されるようになってきた。これらのデバイ

¹ 立命館大学 大学院 情報理工学研究科
Graduate School of Information Science and Engineering,
Ritsumeikan University

² 立命館大学 情報理工学部
College of Information Science and Engineering, Ritsumeikan
University

スから取得できるデータの中には、携帯端末を持ち歩く人間のライフログが含まれていると考えられる。現在、このようなライフログを蓄積・処理することで、持ち主の生活習慣を推測したり、さらにその推測結果から行動予測を行い、NTTドコモのiコンシェルを代表とするレコメンド型サービスなどに活かそうとする研究 [1][2] が行われている。

ライフログを用いた研究を行う際に、研究者は得られたライフログやそのマイニング結果を参照・分析する。ここで、ライフログやマイニング結果を数値や文字の羅列として参照し、データの変遷や、複数データの対応を分析するのは難しい。そのため研究者は参照・分析作業において、データをグラフ化したり、地図上に描画したりする可視化システムを用いるのが一般的である。しかし、現状存在するライフログの可視化システム [3][4][5] では、データを可視化の際に、指定されたデータの期間やサンプル数(レンジ)と、グラフや地図の縮尺(スケール)を考慮した可視化が行えないといった問題がある。たとえば、“特定ユーザの時系列順に並んだ位置情報を地図上に移動軌跡として描画する”といった方法でのデータの可視化を考える。このとき、長いレンジのデータを指定すると、その中には2地点間を何度も同一の経路で往復していたデータや、一箇所で滞在していたデータが含まれる可能性がある。このようなデータをそのまま可視化すると、ユーザの設定したスケールによっては結果が重なり見づらくなってしまふ。さらに、指定したデータのサンプル数が多い場合、データの通信処理やグラフの描画処理に時間がかかり、快適な閲覧が行えなくなってしまうことも考えられる。

そこで本稿では、データマイニングを行うプログラム(以下、マイニングプログラム)の開発において、前述した問題を解決し、ライフログやそのマイニング結果の効果的な閲覧を可能にするシステムである CHANTAKU を提案する。CHANTAKU のユーザは、自身が望むスケール・レンジに合わせたデータの閲覧が可能である。

本稿は全6節で構成されている。第2節では、ライフログを可視化するツールの現状と課題を明らかにする。第3節では、課題に対するアプローチとそれを実現する可視化ツールである CHANTAKU の設計を述べる。第4節では、CHANTAKU の実装について述べ、第5節でその評価を述べる。最後に第6節でまとめと今後について述べる。

2. ライフログ可視化システムの現状と課題

2.1 先行研究

我々の研究室では、ライフログを扱う研究者を対象とした可視化システムである DUBIX の開発を行ってきた [3]。DUBIX は GoogleWebToolkit^{*1}を用いた Web アプリケーションとして実装されており、データを取得する部分とグ

ラフを描画する部分が分離した設計になっているため、多様なフィルタリングに対応した可視化が可能である。ユーザはリストボックスやラジオボタンなどの GUI か、直接クエリを打ち込むことの出来るテキストボックスによって、参照するデータやそのフィルタを指定する。可視化方法は大きく分けて、横軸を時間とした折れ線グラフ(以下、時系列グラフ)と、二次元地図上にデータを描画するグラフ(以下、位置グラフ)の2種類であり、位置グラフには位置情報の箇所にピンを立てる、位置情報同士を線で結んで移動軌跡を描画するなど様々な形態が存在する。複数の可視化方法が存在するデータが指定された場合は、優先度に従って可視化方法が決定されるが、その後可視化方法の切り替えが可能である。また、グラフ上に描画した特定の時刻のデータや、特定の位置でのデータなど、ユーザの見たいある一点(以下、注目点)を動的に指定するためのスライダーが設置されている。DUBIX 上の複数のグラフで同じ時間範囲の複数のデータが可視化されている場合、ユーザはそれぞれの可視化結果の注目点をスライダーによって協調動作させることが可能であり、データの対応を容易に把握できる。この協調動作の様子を図1に示す。さらにグラフの描画位置は、ブラウザの画面範囲内で自由に移動させることができる。



図1 先行研究:DUBIX(スライダーによる協調動作の様子)

2.2 関連研究

2.2.1 位置と画像を関連付けて表示するライフログビューア

仲野ら [4] はデジタルカメラで撮影した写真と、その撮影時刻、姿勢角センサから得られる姿勢情報、GPS から得られた位置情報を一つのライフログとして蓄積・利用するためのシステムを提案している。さらに、そのシステムの一部として、撮影した写真を二次元地図上に表示するライフログビューアを提案している。この研究以前のシステムでは、それぞれの機器から取得した情報をそれぞれの記憶媒

*1 <http://code.google.com/intl/ja/webtoolkit/>

体から取り出し、時刻による関連付けを行うまではデータの参照ができなかった。それに対してこのシステムでは、カメラで撮った写真と姿勢情報・位置情報のマッチング・蓄積を同時に行うことで、即時性を持たせている。ライフログビューアにおいて写真は、それぞれ紐付けられた撮影位置に描画される。ビューアの表示内容は10秒に1度更新されるため、ユーザは即時に、気軽にシステムの利用が可能である。さらに空間把握性を向上させるために、姿勢情報を利用し、実際の撮影方向と一致した画像表示の提案・検討も行っている。

2.2.2 ライフログを構造化して表示する可視化システム

小野ら [5] は、ライフログとしてウェアラブルカメラから得られる画像と音声・加速度情報、さらに腕時計型の高度・脈拍情報取得デバイスから各種情報を取得し、それらの情報を効率的に閲覧するための可視化システムを提案している。この可視化システムでは得られた膨大な情報と、「講義を受けていた」、「研究室で作業を行っていた」などのイベントシーンとをニューラルネットワークによって関連付け、構造化している。ライフログをイベントシーンごとに構造化することによって、単純にライフログを表示するよりもユーザの記憶想起支援を効率的に行える。

2.3 現状のライフログ可視化システムの課題

現状のライフログ可視化ツールでは、指定されたデータのスケールやレンジを考慮した可視化ができない。たとえば、先行研究である DUBIX には、時系列順に並んだ位置情報の可視化手法として、移動軌跡を地図上に描画する方法がある。この可視化手法を用いて、レンジの長いデータを描画した場合、指定されるデータの中には同じ経路の往復を繰り返しているデータや、一定箇所に滞在していたデータが含まれる可能性がある。このようなデータをそのまま可視化した場合、スケールによっては可視化結果が地図上で重なり、閲覧しづらくなってしまふ。2.2.1 項で挙げた可視化システムでは、地図の縮尺によって、写真の描画に姿勢情報を利用するかどうかの検討を行っているが、レンジの増加に伴って写真の量が増えた場合、どのように描画を行うかは考慮されていない。2.2.2 項で挙げた研究では、ライフログの量が膨大になった場合にも、効率的に参照を行うための構造化を可視化システム上で行っている。構造化によって効率的なデータ参照ができるが、データの総量が増えることに変わりはない。したがって、可視化システムによるデータの描画処理における負担が大きくなってしまふ可能性がある。

2.4 要件

ライフログやマイニング結果の閲覧時に、研究者は様々なスケール・レンジでデータを閲覧すると考えられる。その際、すべてのデータを単一の方法・条件で可視化する

と、閲覧しづらくなってしまふ可能性がある。したがって、ユーザがそれぞれのスケール・レンジに合わせた可視化を行えるべきである。さらに、このような可視化に際して、2.3 節の最後で述べたような可視化システムへの負担増加を極力抑えられることが望ましい。

3. CHANTAKU の設計

3.1 アプローチ

2.4 節で述べたスケールやレンジ変化への対応の方法として、2.2.2 項で挙げた研究が行っていたようなデータの構造化が考えられる。しかし、データの構造化のみであると、可視化対象データが膨大になった場合に、システム上で行うデータの送信や描画処理の負担が増加してしまふ。

そこで、本研究では可視化システム上でスケール・レンジに合わせたデータの間引き処理を行う。間引き処理には、その程度を決定する閾値が用いられるが、これらの閾値を指定されたデータのレンジと、可視化結果のスケールによってシステムが自動で決定する。この間引き処理を行うことで、ユーザはスケール・レンジの変化に合わせたデータの分析を行うことが可能であり、同時にデータのサンプル数が減るため、システムの描画処理による負担も軽減されると考えられる。また、可視化システム上で間引き処理を行えるようにすることで、ユーザが事前に間引きを施したデータを用意する負担を減らすことができる。

さらに、このような間引き処理はユーザによって許容できる度合いが異なるため、1度可視化された後、間引き処理に用いる閾値を動的に変更可能にする。また、この閾値の動的な変更を快適に行うために、ユーザが参照する可能性の高い処理結果をプリフェッチする機構を設ける。

3.2 設計

3.2.1 システム概要

CHANTAKU のシステム概要図を図2に示す。CHANTAKU は大きく分けてデータ取得部とグラフ描画部の2つの構成要素から成り、それぞれがサーバ、クライアントと対応している。以下でグラフ生成までの処理の流れを述べる。なお、図2では説明のため、DBMS:A を参照するパターンで間引き処理なしの場合、DBMS:B を参照するパターンで間引き処理ありの場合の処理をそれぞれ表している。

- (1)-(3) データ取得モジュール判定部では、データ指定・グラフ描画パネルでユーザが指定したデータベース情報から、利用するデータ取得モジュールを決定する。データ取得モジュールでは、フィルタ情報から対応する DBMS に合わせたクエリを生成し、実行する。
- (4)-(5) データベースから取得したデータは、DBMS の種類によらず、すべて可視化対象データという共通の形式に変換して、データ格納部に格納する。ここで

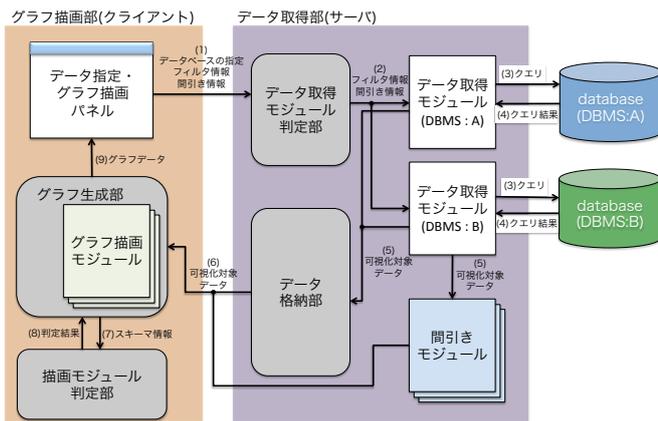


図 2 システム概要図

ユーザが間引き情報を指定していた場合、間引き処理を行う。

(6)-(9) 描画モジュール判定部では、データ取得部から送られてきた可視化対象データの含むスキーマ情報を基にグラフ形式を判定する。ここで判定されたグラフ描画モジュールを用いてグラフ生成部ではグラフデータを生成し、データ指定・グラフ描画パネルでグラフの描画を行う。

3.2.2 間引き処理に使う閾値の動的変更機能

図 3 に、間引き処理に使う閾値の動的変更機能の処理フローを示す。まず、ユーザが閾値を変更すると、システムは変更後の閾値を用いた間引き処理結果がキャッシュとして存在するか確認する。キャッシュが存在した場合は、それを用いてデータの可視化を行う。存在しなかった場合は、変更後の閾値を用いて間引き処理を行い、処理結果を可視化する。可視化処理の終了後、システムはプリフェッチアルゴリズムに従い、プリフェッチを行う必要があるか判別する。ここで、必要があれば再び間引き処理を行い、結果をキャッシュして処理を完了する。

3.2.3 間引きモジュールの設計

膨大なデータを間引く方法としては、対象データの種類によって様々な方法が挙げられる。このようなデータの種類と対応する間引き処理の一例を表 1 に示す。

表 1 データの種類と対応する間引き処理の例

データの種類	対応する間引き処理
時刻をキーとして保存された位置情報	Ramer Douglas Peucker algorithm
位置情報、時刻をキーとして保存されたスカラ	クラスタリング
時刻をキーとして保存されたデータ	時間ごとに間引く処理
共通	サンプル数ごとに間引く処理

例えば、時系列順に並んだスカラは、クラスタリングによって間引くことができる。また、位置情報であれば、時

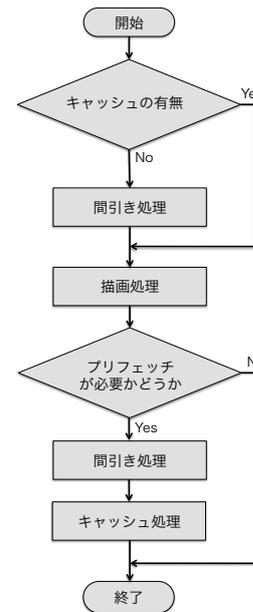


図 3 閾値動的変更機能の処理フロー

系列順に並んだ位置情報を対象とし、移動軌跡をスケールに合わせて簡素化するための Ramer Douglas Peucker アルゴリズムなどが存在する。

しかし、1 節で挙げたような、同一経路での移動や、同一箇所での滞在により重複した位置情報を、スケール・レンジ両方に合わせて効果的に間引く方法は存在しない。そこで、本研究では間引きモジュールの一例として、携帯端末から得られる、時系列順に並んだ位置情報の中に、先に挙げたようなデータが含まれていた場合間引き処理を行い、滞在していた地点と経路を抽出するモジュールを設計・実装する。この間引きモジュールでは、指定されたデータのレンジが極度に長い場合でも、ユーザが快適に可視化結果を閲覧できるように、間引き処理に用いる閾値をレンジに応じて決定する。この間引きモジュールの利用法としては、初めて見るユーザの行動特徴の把握などが挙げられる。以下にこのモジュールで利用するアルゴリズムの詳細を述べる。

屋内の滞在地抽出アルゴリズム

ユーザの滞在地は屋内・屋外などの条件や滞在の定義によって異なる。そこで、本稿では滞在地抽出の第一段階として GPS の取得できていない、ユーザが屋内にいたと考えられる滞在地の抽出・登録を行う。この滞在地を抽出・登録するアルゴリズムでは、時間閾値、距離閾値、速度閾値、精度閾値、同一滞在地判定閾値の 5 つの閾値を用いる。本稿では、GPS から得られる位置情報の取得間隔が時間閾値を超えていた場合、その間“GPS が消失していた”と定義する。このとき、前の位置情報の地点で“GPS が消失した”，後の位置情報の地点で“GPS が復活した”と定義する。図 4 に滞在地抽出・登録処理フローを示す。

(1) GPS の消失をシステムが検出すると滞在地抽出判定

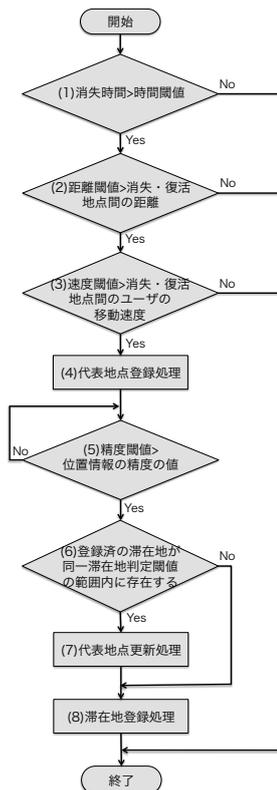


図 4 滞在地抽出・登録処理フロー

が行われる。

- (2)(3) 滞在地抽出判定では消失・復活地点間の距離と、消失・復活地点間のユーザの平均移動速度を距離閾値、速度閾値とそれぞれ比較する。このとき、消失・復活地点間の距離が距離閾値を超えず、さらに消失・復活地点間のユーザの平均移動速度が速度閾値を超えていなかった場合、ユーザは屋内に停留していたと見なし、滞在地の抽出を開始する。
- (4) 滞在地の抽出開始時には、GPS が消失した地点を代表点として登録する。
- (5) 滞在地の抽出処理の終了判定は、GPS から得られる位置情報の精度の値を基に行う。精度の値は小さければ小さいほど、精度が高いことを意味する。したがって、精度の値が精度閾値を下回った場合、ユーザは屋外に出た可能性が高いと判断し、滞在地の抽出を終了する。
- (6)-(8) 抽出した滞在地の代表地点から、同一滞在地判定閾値の範囲内に登録済みの滞在地が存在した場合、2つを同一の滞在地と見なし、抽出した滞在地の代表地点を登録済みの滞在地に更新する。

経路抽出アルゴリズム

本稿では、滞在地間で取得できる位置情報を“経路データ”，さらに出発時刻・到着時刻と出発地，到着地，経路データに対応する識別子を持つデータを“経路情報”と定義する。ここで経路データは、同一のものであれば間引く

ことが可能である。そのため、まず前節で述べたような滞在地の抽出に合わせて、得られる経路データと経路情報も抽出し、その後、どの経路データを登録するかを判定を行う。この登録経路判定処理フローを図 5 に示す。

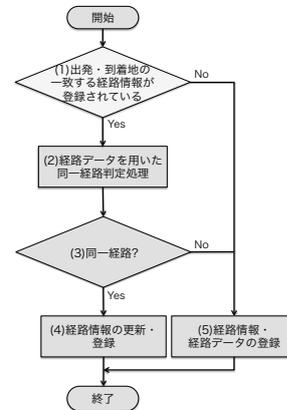


図 5 登録経路判定処理フロー

- (1) 現在注目している経路の経路情報と、登録済みの経路情報を比較し、出発地・到着地が一致しているかどうか確認する。
- (2)(3) 出発・到着地の一致する経路情報が存在した場合、それぞれ対応する経路データを用いて同一経路判定処理を行う。
- (4) ここで、同一経路と判定された場合は、現在注目している経路情報の持つ経路データ識別子を登録済みのものに更新し、経路データは登録しない。
- (5) (1) で出発地・到着地が一致しなかった、あるいは同一経路判定処理で、同一の経路ではないと判定された場合には、そのまま経路情報と経路データの登録を行う。

なお、本稿では同一経路判定処理に参考文献 [1] で提案されている手法を利用する。

4. CHANTAKU の実装

CHANTAKU は先行研究である DUBIX の拡張として実装した。本節では間引き処理に関する CHANTAKU の実装について説明する。

4.1 間引き処理結果用描画モジュールのプロトタイプ実装

3.2.3 項で述べた方法で間引いた結果を描画するモジュールは Google Maps Library for GWT 1.1.0^{*2}を用いて実装を行った。この描画モジュールを用いる際、指定するデータは時系列順で GPS から得られる位置情報である。この描画モジュールでは、滞在していたと判定された区間の代表地点に、それぞれの滞在時間に応じた色のマーカーを描

^{*2} <http://code.google.com/p/gwt-google-maps/wiki/MapsGettingStarted>

画する。また、経路データを通過回数に応じた色のポリラインで描画する。このモジュールを用いて可視化を行った例を図6に示す。



図6 間引き処理結果の可視化

4.2 スケール・レンジによる間引き処理の閾値変更

本稿では、3.2.3項で述べた間引きモジュールと、4.1節で述べた間引き結果用描画モジュールに対応する形で、指定されたレンジによって時間閾値を、地図のスケールによって同一滞在地判定閾値を変更する機構を設けた。この機構は、他の間引きモジュール、描画モジュールに再利用が可能である。閾値が変更された際の、再度間引き処理を行うプロセスを図7に示す。図7において、赤い矢印がキャッシュがクライアント側に存在する場合のプロセス、黒い矢印がキャッシュが存在しない場合のプロセスである。

キャッシュが存在する場合、システムはそのキャッシュとグラフ描画モジュールを用いてグラフデータを生成し、描画する。キャッシュが存在しない場合、システムはクライアント側で変更された閾値情報をサーバに送る。サーバでは、3.2.1項で述べた、データストアから取得した後格納しておいた可視化対象データと、送られてきた閾値情報を用いて間引き処理を行う。間引き処理結果は再びクライアントのグラフ生成部に送られ、グラフ生成部ではそれを基にグラフデータを生成する。

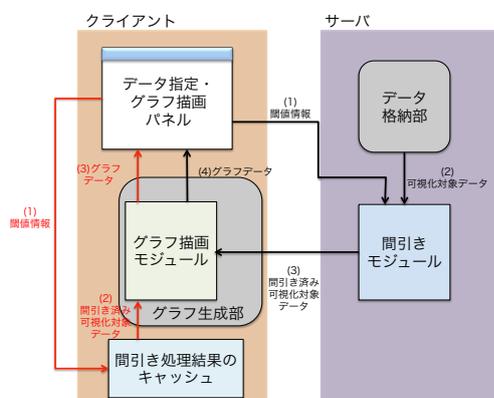


図7 間引き処理結果の再取得プロセス

レンジを考慮した可視化

レンジが1日程度のデータを見る際には、電車の停車や信号待ちなどによる短い時間の滞在は十分に参照される可能性がある。しかし、1週間、1ヶ月といった長いレンジのデータを見る際に、このような短い時間の滞在が参照されるとは考えにくい。そこで、レンジに応じて時間閾値を自動的に変更する機構を設けた。図8にこのレンジを考慮した閾値変更の有無による可視化結果の変化を示す。なお、図8では2週間分のデータを間引いて可視化している。



図8 レンジを考慮した閾値変更の有無による可視化結果の変化

スケールを考慮した可視化

可視化結果のスケールによって間引き処理結果は変更する必要がある。4.1節で述べたような地図上にデータを描画する可視化方法において、縮尺が大きくなった場合、ユーザは指定したデータを詳細に閲覧しようとしていると考えられる。そこで、この間引き処理結果描画モジュールにおいて縮尺が大きくなった場合、同一滞在地判定閾値を変更し、間引かれていた結果が表示されるようにした。図9にスケールを考慮した閾値変更の有無による可視化結果の変化をそれぞれ示す。図9において、左側がスケールの変化を考慮しなかった場合で、右側がスケールの変化に応じて閾値の変更を行った場合の可視化結果である。

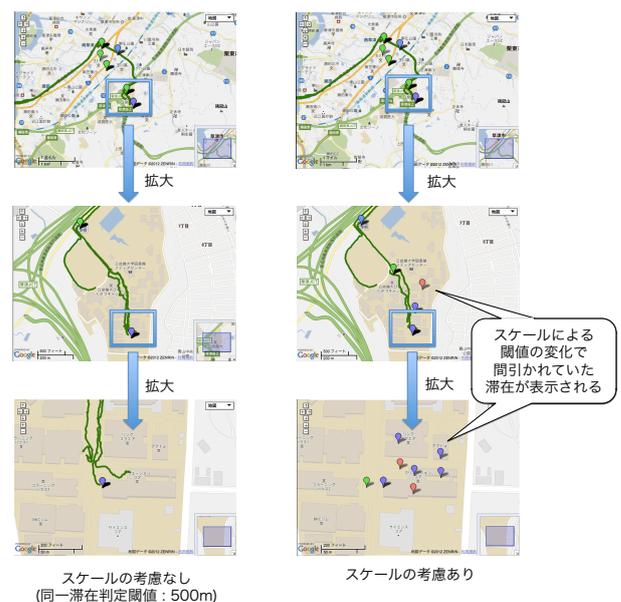


図9 スケールによる可視化結果の変化

4.3 スライダーによる間引き処理の閾値変更

DUBIX には動的な注目点移動のためのスライダーが設置されている。CHANTAKU ではそれを閾値の変更にも利用する。スライダーに登録する閾値・注目点の切り替えには、描画グラフの下に設けられたリストボックスを用いる。リストボックスによる閾値・注目点切り替えの様子を図 10 に、スライダーを用いた動的な閾値変更の様子を図 11 に示す。図 10 でスライダーに登録した閾値は、図 4 で示した滞在地登録処理で用いる同一滞在地判定閾値である。この閾値を小さくすることで、滞在地は同一と判定されづらくなるため、マーカーで描画される滞在地の数は増加する。



図 10 リストボックスによる閾値・注目点切り替え



図 11 スライダーによる動的な閾値変更

4.4 処理結果のプリフェッチ機能

スケール・レンジの変化を快適に行うため、現在の各閾値を用いて、ユーザが参照する可能性の高い処理結果をプリフェッチする機能を実装した。この機能に利用するプリフェッチアルゴリズムとして、現在のスライダーの位置から前後 N 箇所分の処理結果をプリフェッチするという単純なアルゴリズムの実装を行った。

5. 評価

本稿では、提案・実装を行ったシステムに関して、2.4 節で挙げた要件を満たしているかを定性評価で確認する。また、4.4 節で述べたプリフェッチ機能に関して、間引き処理結果をプリフェッチすることで、どの程度処理速度が向上するかを定量評価で確認する。

5.1 定性評価

図 8, 図 9 で示したように、可視化対象データのレンジや、可視化結果のスケールの変化に合わせた適切な閾値をシステムが自動で判断し、間引き処理を行うような機能を実現した。図 8 から、指定されたレンジと比較して期間が極端に短い滞在が間引かれていることがわかる。このような間引きによって、レンジの長さによって可視化結果が重なる部分が減り、見易くなると考えられる。

一方図 9 では、スケールによる閾値変化の有無による可視化結果の違いを示した。図 9 の左側のように、スケールによる閾値変化がない場合、ユーザはデータの詳細を確認しようとした場合、自身で閾値の変更をし、スケールにあった可視化を行う必要がある。しかし、図の右側のように、自動的にスケールに対応した閾値変化をすることで、ユーザが自身でスケールに合わせて閾値を変更する負担が減ったと考えられる。

さらに、それらの間引きがユーザに好ましくないものであった場合、4.3 節で述べたように、間引きに用いる閾値をスライダーによって動的に変更できるような機能も実装した。これらの実装により、ユーザは望むスケールやレンジに合わせて、データ閲覧をしやすい適切な可視化を行えるようになったと考えられる。

しかし、今回実装を行った間引きモジュールでは、間引く前のデータと間引いた後のデータそれぞれで利用するグラフ描画モジュールが異なったため、2.4 節の最後で述べたような、データの描画処理にかかる負担軽減が実現できているかどうかは評価することができなかった。

5.2 定量評価

本稿では定量評価として、4.4 節で述べた間引き処理結果をプリフェッチする機能の有用性を評価する。この定量評価に用いたサーバの詳細を表 2 に、クライアントの詳細を表 3 にそれぞれ示す。

表 2 評価に用いたサーバの詳細

CPU	Intel Core i7 3.40GHz
OS	Ubuntu 10.04.3 LTS
メモリ	4GB
Web コンテナ	Apache Tomcat 6.0

表 3 評価に用いたクライアントの詳細

CPU	Intel Core i7 1.8GHz
OS	Mac OS X 10.7.2
メモリ	4GB
ブラウザ	Google Chrome16.0.912.77

プリフェッチ機能の有用性は、間引きの手法やその描画方法、プリフェッチアルゴリズムなどの要因によって異なる。しかし、どのような間引き処理とその描画処理を利用

する場合でも 4.2 節で述べたようなプロセスを経ることに変わりはない。そこで本稿では有用性の指標として、実装した間引きモジュールを用いて、キャッシュ機構があり、且つキャッシュが存在する場合と、キャッシュ機構がない場合で処理時間にどの程度差が出るかを計測し、評価を行う。この評価には、あるユーザの GPS ログ 1 日分と 1 週間分を用い、閾値の変更は同一滞在地点判定閾値を 500m から 100m に変更するという処理を行った。処理時間はデータの期間と、キャッシュの有無の組み合わせでそれぞれ 10 回ずつ測定を行い、平均を算出した。評価に用いた各閾値を表 4 に、データのサンプル数と間引き処理結果の詳細を表 5 に示す。また、処理時間の測定結果を表 6 に示す。

表 6 より、測定結果は 1 日分のデータ、1 週間分のデータどちらの間引き処理に関しても、キャッシュをしていた場合の方が明らかに処理時間が短かった。これは 3.2.2 項で述べたように、キャッシュを保持していた場合に要する処理時間が、クライアントでのデータの参照・描画処理のみであり、キャッシュを保持していない場合に要する、サーバでの間引き処理時間や、通信処理にかかる時間が軽減されているからであると考えられる。この結果より、クライアントに間引き処理結果をキャッシュする機能が、本システムにおいて有用であるといえる。

表 4 評価に利用した間引き処理の閾値

閾値の種類	1 日分	1 週間分
時間閾値	30 秒	60 秒
距離閾値		100m
速度閾値		1.53m/s
同一滞在地点判定閾値		100m

表 5 評価に用いたデータと間引き結果の詳細

	サンプル数	抽出滞在地点数	抽出経路数
1 日	4757	11	12
1 週間	41285	31	155

表 6 測定した処理時間

	1 日	1 週間
キャッシュなし	336.6ms	4226.3ms
キャッシュあり	220.3ms	916.1ms

6. 結論

6.1 本研究のまとめ

本稿では、ライフログやそのマイニング結果を参照・分析するための可視化ツールの現状を説明し、スケール・レンジを考慮したデータ閲覧ができない問題が存在することを明らかにした。さらにその問題を、間引き処理によって解決する可視化ツールである CHANTAKU を提案・実

装し、評価を行った。その結果、上記の問題が解決され、CHANTAKU のユーザはスケール・レンジの考慮された可視化を行うことができるようになった。さらに、スケール・レンジの変化に素早く対応するため設けた間引き処理結果のプリフェッチ機能に関して、キャッシュの有無による処理速度の違いを計測し、その有用性を示した。

6.2 今後の課題

間引きモジュールの開発・充実

本稿において実装した間引きモジュールは、ユーザの屋外での滞在は検知できない。また、速度閾値を GPS が消失する直前の速度に合わせるなどの、相対的な閾値の変化が可能であると考えられるが、実現できていない。さらに、プリフェッチ機能も単純に前後 N 箇所の閾値を用いた間引き処理結果をプリフェッチするものしか実装されていない。したがって、本稿で実装した間引きモジュールやプリフェッチ機能は引き続き開発・検討を行い、より有用にする必要がある。さらに、今回実装した間引きモジュール以外にも、CHANTAKU には様々な間引きモジュールが適応可能である。このような間引きモジュールを充実させることで、より多くのユーザに対応した可視化ツールを実現できるだろう。

インタフェースの改善

本稿では、間引きモジュールの閾値や、グラフの注目点を変更するためのインタフェースとして、リストボックスと、先行研究の DUBIX に設けられていたスライドバー 1 つを採用した。しかし、今回実装を行った間引きモジュールのように、複数の閾値を変更する可能性がある場合に、現在のインタフェースが適当だとは言いがたい。したがって、より良いインタフェースの考案・実装を行う必要がある。

参考文献

- [1] 山田直治, 磯田佳徳, 南正輝, 森川博之. 屋外行動支援のための GPS 搭載携帯電話を用いた移動経路の逐次的精練手法 情報処理学会論文誌, Vol52, No.6, pp.1951-1967, 2011
- [2] 西野正彬, 中村幸博, 武藤伸洋, 阿部匡伸. あいまいな表現を含むスケジューラデータと GPS データとの時間的共起関係を利用した行動予測手法の検討 電子情報通信学会研究報告 第 109 巻, pp.73-78, 2010
- [3] 浜地亮輔, 北沢匠, 榎堀優, 新井イスマイル, 西尾信彦. 効果的なライフログ閲覧のための柔軟な配置と協調表示可能な可視化システム. マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO2011). 2011.
- [4] 仲野潤一, 西村邦裕, 谷川智洋, 廣瀬通孝. 位置と画像を利用したリアルタイムライフログシステム, 情報処理学会第 70 回全国大会, CD-ROM, pp227-228, 2008
- [5] 小野将之, 西村邦裕, 谷川智洋, 廣瀬通孝. 多様なセンサによるライフログのニューラルネットワークを用いた構造化, 電子情報通信学会研究報告 第 109 巻, pp79-84, 2009