

# モバイルセンシングデータ可視化のための ライフログ可視化フレームワーク MashMap の拡張

高橋 昂平<sup>†</sup> 佐伯 幸郎<sup>†</sup> 梶本 真佑<sup>†</sup> 中村 匡秀<sup>†</sup>

<sup>†</sup> 神戸大学 〒 657-8501 神戸市灘区六甲台町 1-1

E-mail: <sup>†</sup>koupe@ws.cs.kobe-u.ac.jp, <sup>††</sup>sachio@carp.kobe-u.ac.jp, <sup>†††</sup>{shinsuke,masa-n}@cs.kobe-u.ac.jp

**あらまし** モバイルセンシングは、ユーザがセンサを持ち歩いて自律的に動き、その周囲の状況をセンシングするユーザ参加型センシングの手段である。モバイルセンシングのデータを閲覧・分析するには地図上に可視化することが効果的である。我々は、位置情報が含まれる様々なライフログをマッシュアップし、地図上に可視化するフレームワーク MashMap を提案している。本稿では、MashMap を拡張して、モバイルセンシングデータを地図上に可視化できる枠組みを開発する。提案手法では、モバイルセンシングデータをライフログの一つとみなし、MashMap が扱うライフログの標準形式に変換する方法を考察する。また、センシングデータを効果的に可視化するために、同系列のデータを計測値に応じて表示形式を変える機能や、計測値によってデータの絞り込みを行う機能を追加実装する。ケーススタディとして、速度センサのログを可視化し、提案フレームワークの適用可能性を考察する。

**キーワード** モバイルセンシング, ライフログ, マッシュアップ, 位置情報, フレームワーク

## Extending Lifelog Mashup Framework MashMap for Mobile Sensor Data Visualization

Kohei TAKAHASHI<sup>†</sup>, Sachio SAIKI<sup>†</sup>, Shinsuke MATSUMOTO<sup>†</sup>, and Masahide NAKAMURA<sup>†</sup>

<sup>†</sup> Kobe University Rokko-dai-cho 1-1, Nada-ku, Kobe, Hyogo, 657-8501 Japan

E-mail: <sup>†</sup>koupe@ws.cs.kobe-u.ac.jp, <sup>††</sup>sachio@carp.kobe-u.ac.jp, <sup>†††</sup>{shinsuke,masa-n}@cs.kobe-u.ac.jp

**Abstract** Mobile sensing is a means of participatory sensing where mobile users carry hand-held sensors and measure user's surroundings. The map-based visualization is useful to see and analyze the mobile sensing data. We have previously proposed *Mashmap*, which is an application framework supporting developers to visualize location-based lifelogs. In this paper, we develop a new framework for map-based visualization of the mobile sensing data, by extending the MashMap framework. The proposed framework regards the mobile sensing data as one of location-based lifelogs, and adapts the sensing data to the lifelog common data model of MashMap. For effective visualization, we implement a feature that can define value-based display format of data items within a single data source. We also implement a feature of nested filters for flexible extraction of interesting data. A case study with the velocity sensor data demonstrates the practical feasibility of the proposed framework.

**Key words** mobile sensing, lifelog, mashup, location data, framework

### 1. はじめに

スマートフォンの普及やセンサデバイスの小型化・低価格化に伴い、人間がセンサを持ち歩き、自律的に移動しながら自らの周囲の環境や状況を容易に測定できるようになってきている。このように、ユーザがセンサを持ち歩き、移動しながらその周囲の状況をセンシングすることを、本稿では**モバイルセンシング**と呼ぶ。モバイルセンシングを共通の目的の下で多数のユーザで実施し、大量のデータを集める活動は**ユーザ参加型センシ**

**ング** [1][2] と呼ばれ、近年注目を集めている。

モバイルセンシングによって収集されるデータにはその性質上、位置情報が含まれる。特に、同じ種類のデータを様々な地点で取得した場合、地図上での可視化によってデータを俯瞰的に見ることができ、非常に有効である。しかしながら、一般にこうしたデータの可視化は、目的や用途ごとに専用のプログラムを作成する [3] ことが一般的であり、アプリケーションやサービスの開発を複雑化させる一因となっている。

我々は先行研究 [4] [5] において、様々な種類の位置付きライフログデータを地図上に可視化するための汎用的なアプリケーションフレームワーク *MashMap* を提案している。MashMap では、様々な形式の位置付きライフログを、我々の研究グループで提案しているライフログ標準データモデル (LLCDM) 形式 [6] [7] に変換してデータベース (DB) に蓄積する。アプリケーション開発者は、必要なデータを DB から選別するためのフィルタとその表示方法を定義し、データソースを作成する。次に、定義したデータソースを一つあるいは複数選択して、MashMap を定義する。定義された MashMap は最終的に Google Map 上に可視化される。MashMap フレームワークを用いることで、開発者は地図を用いた様々なライフログサービスを迅速かつ容易に開発できる。

本研究では、MashMap を拡張して、ライフログのみならず、モバイルセンシングデータを地図上に可視化できる新たな枠組みを開発する。提案手法では、まずモバイルセンシングデータをライフログの一つとみなし、MashMap が扱うライフログの標準形式 LLCDM に変換する方法を考察する。具体的には、モバイルセンシングで測定したデータの属性を、5W1H(When, Where, Who, What, Why, How) 観点から分類し、LLCDM の各属性にマッピングする。また従来の MashMap では、一つのデータソースに対して一つの表示方法しか定義できず、センシングデータの効果的な可視化に限界があった。提案手法では、同じデータソース内のデータに対して、測定値に応じて異なる表示方法を定義できる機能を実装する。さらに、アプリケーションがより柔軟なデータの選別を行えるよう、計測値に応じたフィルタを複数組み合わせられる機能も追加実装する。

開発した新たなフレームワークの実用性を評価するため、ケーススタディを行った。モバイルセンシングによって、旅行中、移動する車内で取得した速度センサのログデータを可視化した。その結果、どの地点をどれくらいの速度で走っていたかを詳細に振り返る地図を容易に作成でき、提案フレームワークの有用性が確認できた。

## 2. 準備

### 2.1 モバイルセンシング

本論文において、モバイルセンシングとは、ある位置に固定した特定の空間の情報を特定のセンサーによって収集するのではなく、人間が自立的に動くことで、そのユーザの周囲の状況をセンシングすることを指す。取得するのは、外気温、湿度、気圧、騒音、明るさなどの環境データの他、ユーザ自身の判断した「天気が晴れである」、「ここは混んでいる」といった情報もセンシングデータとなりうる。この他、加速度や地磁気、GPS 等のセンサは昨今のスマートフォンには標準搭載されており、非常に簡単にモバイルセンシングを行うことができる。データには、その測定値のメタデータとして取得日時や取得地点、測定者、測定機器などの情報が付加されるのが一般的である。また、ユーザ各自が携帯端末を持ち歩くことにより、広範囲の情報を複数ユーザの複数センサから収集することが可能と



図1 参加型センシングによる情報を利用したサービスの例

なっている [8]。このデータを利用するユーザ参加型のセンシングといった手法も注目されている。

これを利用した例として、図 1(a) は、ウェザーニュース社のウェザリポートチャンネル [9] の画面である。このサービスでは、ユーザがレポートしたその地点のピンポイントな天気を同一の地図上に可視化し共有できるサービスである。また図 1(b) は、みんなで作る放射線量マップ [10] の画面である。福島原発事故後に開発されたこのサービスは、ユーザがそれぞれ自ら放射線量を測定した結果を集約して表示し、広域の放射線量マップを作成することを実現している。

### 2.2 ライフログのマッシュアップ

人間の日々の行動をデジタルデータとして記録するライフログが注目されており、Web 上には様々なサービスが存在する。代表的なものとして、つぶやきを記録・共有する Twitter [11] や、写真を保存・共有する Flickr [12] などがある。様々なサービスで記録された複数のライフログを統合・集約することで、より高い付加価値を持つサービスへと発展させることができる。

こういったライフログの高度かつ柔軟なマッシュアップを支援するため、我々は先行研究 [6] [7] において、ライフログのための標準データモデル (Life Log Common Data Model, LLCDM) およびマッシュアップ API (Life Log API, LLAPI) を提案している。

LLCDM は 5W1H の観点からライフログが備えるべきデータ項目を整理し、特定のサービス・アプリケーションに依存しないスキーマを定義している。また、アプリケーションに依存するログの内容に関しては、データ自体や構造は保持するが、自らは解釈せず、外部のスキーマに任せるといった構造をとっている。

LLAPI は LLCDM 形式に変換されたライフログデータにアクセスするための標準的なインタフェースを定義する。LLCDM の各データ項目に対してクエリをかけ、必要なライフログデータを取得することができる。これにより様々な種類のライフログへ統一的にアクセスすることができる。

## 3. 先行研究：MashMap フレームワーク [4] [5]

### 3.1 概要

MashMap フレームワークは、様々な形式の位置付きライフログを地図上に可視化するためのアプリケーションフレームワークである。“MashMap”とは、様々な位置付きライフログを

マッシュアップ (Mashup) して地図 (Map) に表示するという意味を含めた造語である。フレームワークにおいて開発者は、位置付きライフログから自分の必要なデータを選んでデータの源泉 (データソース, data source) とし、複数のデータソースをマッシュアップした地図を作成する。この地図を MashMap と呼ぶ。

図 2 にフレームワークの全体像を示す。図の最上部は、ユーザーが様々なサービスで記録した位置付きライフログを表している。これらの様々な形式の位置付きライフログのデータを、提案フレームワークのデータベース (DB) へ随時インポートしておく。この際に、2.2 節で述べた、LLCDM 形式に適宜変換する。これにより、LLAPI を介して様々な種類へ統一的にアクセスすることができる。

アプリケーションの開発者は、DB に蓄積されたライフログデータから自分が必要なものを選別して、データソースを作成する。データソース作成においては、データの選別に用いるフィルタ (Filter) と、選別したデータをどのような形式で地図上に表示するかを定義する表示形式 (Display Format) を指定する。

次に開発者は、MashMap オブジェクトを作成し、作成済みのデータソースに接続する。異なるデータソースを複数組み合わせさせて接続することが可能である。こうして作成された MashMap オブジェクトは、MashMap Renderer によって Google Map 上に可視化される。地図上では、MashMap 内の複数のデータソースが、それぞれ指定された表示形式で描画され、複数の位置付きライフログがマッシュアップされた地図ができ上がる。

MashMap の作成やデータソースの設定などは MashMap API を通じて行うことができる。MashMap API は MashMap フレームワークの各種オブジェクトを操作するためのインタフェースを提供するファサードクラスである。Filter や DisplayFormat, MashMap の各種オブジェクトの生成、消去、更新を行う。また、データソースの一覧を取得したり、データソースからデータ系列を取得したりすることができる。

また、フレームワークのコアの部分 (図 2 中央四角部分) をクラウド上のサーバに配置して、MashMap フレームワークをサービスとして使う (Software as a Service, SaaS) ことも可能である。

MashMap Builder は、MashMap API の開発者向け GUI フロントエンドである。画面上で Filter, DisplayFormat, DataSource をそれぞれ定義・選択し、新たな MashMap オブジェクトを簡単かつ迅速に作成することができる。

### 3.2 モバイルセンシングへの適用可能性

モバイルセンシングによって得られるデータは、「モバイル」という言葉の示す通り、原則として観測地点の位置情報を持つ。データの種類には依るが、モバイルセンシングによるデータを地図上で可視化したい場合、その処理自体は位置付きライフログを利用したサービスにおいても、センシングデータの可視化においても同様である。特に同じ種類のデータを様々な地点で収集した場合には、地図での可視化によってデータを俯瞰でき、分析や考察に有効である。しかし、一般的にこのようなセンサデータの可視化は、その目的や用途ごとに専用のプログラムを作成することが多く、アプリケーションやサービス開発を複雑

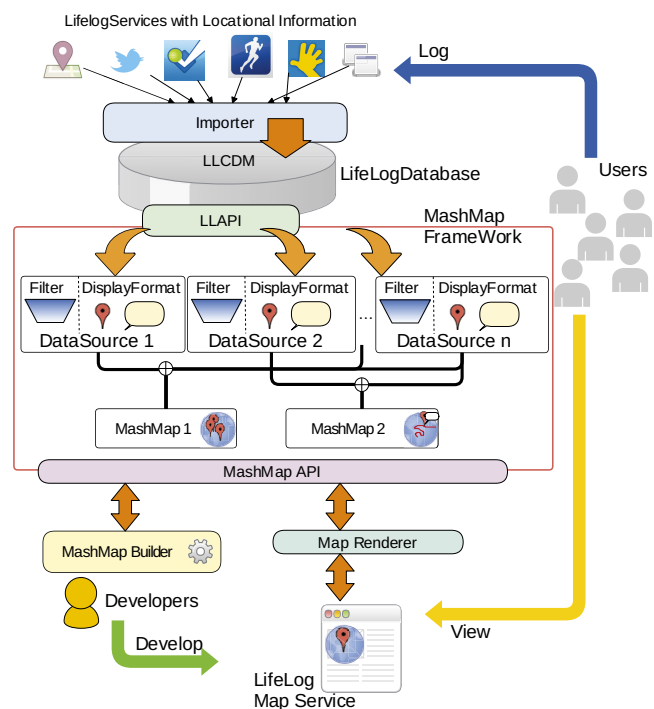


図 2 MashMap フレームワークの全体図

化させる一因となっている。

そこで、MashMap フレームワークを利用して、モバイルセンシングデータを可視化することを考える。提案フレームワーク内では、LLCDM 形式に沿ったログデータであれば統一的に扱えるため、データベースに取り込み利用することができる。よって、モバイルセンシングによって収集したデータを提案フレームワークに取り込むことができれば、センシングデータ同士のマッシュアップや、さらにはライフログとのマッシュアップも可能となる。

しかし、データを現状のフレームワーク内で扱うことができるだけでは、効果的な可視化を行うには不十分である。センシングによって収集したデータを多角度から分析するためには、どのようなデータを選別するのか細かく変更できたり、様々な選別法を組み替えたりすることや、測定値の傾向などを地図上で一覧できる必要がある。

## 4. MashMap フレームワークのモバイルセンシングデータ可視化に向けた拡張

### 4.1 目的とアプローチ

したがって本稿では、次の R1~R3 の要件を満たすように MashMap フレームワークの拡張を行うことを目的とする。

**R1:** モバイルセンサをライフログの一つとして管理する  
MashMap フレームワークは、あくまでも位置付きライフログの可視化を目的としているため、そのままの状態ではフレームワークに取り込むことはできない。また、その可視化に適した機能を現状で備えているとはいえない。しかし、提案フレームワーク内では、ライフログを 2.2 節で述べた LLCDM 形式という共通の形式で管理しているため、この形式に変換すれば、フレームワーク内でセンシングデータを取り扱うこと自体は可能

になる。

### R2: 様々な角度からデータの抽出を行う

センサデータは、そのデータの種類や、可視化によって得たい情報によって様々な取り出し方が考えられる。例えばあるユーザの移動中の速度であれば、その移動をしていた日時によって絞り込むが、ある地域の混み具合を見るときには時間帯と地域によって絞り込みを行い、さらに時間帯毎に比較するといったこともするであろう。そのため、柔軟なデータ抽出をサポートするとともに、それらを様々な組み合わせで利用できるとよい。

### R3: データ値に応じて地図上での表示方法を変える

センサデータを可視化する上で、その測定値は非常に重要であり、表示されているデータの値や傾向をひと目で確認できることが望ましい。例えば値が一定の範囲の場合に色や形を設定して可視化するなど、センサ値によって表現を変える必要がある。そうでなければ「どこでデータを取得したか」以上の情報が得にくいであろう。

以上を考慮した上で、MashMap フレームワークの機能を拡張し、センシングデータに適した柔軟なデータの選別及び可視化を行えるようにする。具体的には、以下のアプローチで拡張を行う。

#### A1: モバイルセンシングデータの LLCDM 適用

#### A2: Filter の拡張

#### A3: DisplayFormat の拡張

詳しくは次節移行で順に説明する。

### 4.2 A1: モバイルセンシングデータの LLCDM の適用

まず R1 を満たすため、センシングデータをライフログの一種とみなすことで提案フレームワークに取り込む。ライフログのための標準データモデル (LLCDM) に変換できれば、フレーム内で扱うことは可能になる。2.1 節で触れたように、センシングデータには、センサの取得データの他にメタデータが付加されている。LLCDM ではデータを 5W1H の観点でデータ項目を定義しており、When には取得日時、Where には取得地点、Who に測定者、What に測定したデータ、How には測定機器の情報が分類できる。また、LLCDM ではデータを共通の形式で取り扱うことで様々な用法で利用しようという思想があるため、Why の観点には触れない。

このように分類した結果から逆に見ると、センシングデータは、「測定者がその日時・その地点において、あるセンサデータを取得した」という人間の行動とみなすことができるため、一種のライフログであると捉えることができる。

この観点でデータを LLCDM 形式に変換し、フレームワーク内に取り込むことで、位置付きライフログだけではなく、位置情報を含むセンシングデータも含めた地図への可視化サービス開発の支援が可能になる。さらに、ライフログとセンシングデータのマッシュアップも可能となる。

### 4.3 A2: Filter の拡張

次に、R2 を達成するために Filter を拡張する。3.1 節でも述べたように、Filter は様々な位置付きライフログが蓄積された DB からデータを選別する役割を持つ。

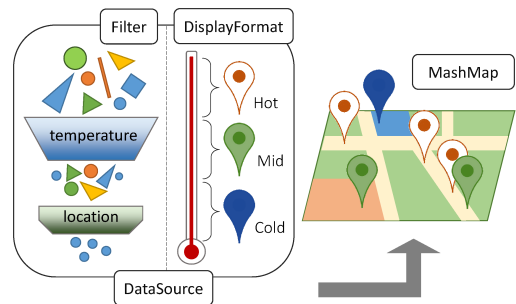


図3 拡張した Filter, DisplayFormat による可視化

これを図3に示すように拡張する。図3は、拡張した Filter 及び DisplayFormat から定義された DataSource が可視化される仕組みを模式的に表した図である。Filter は図の最も左側である。

既存手法においては、DataSource は必ず1つの Filter と1つの DisplayFormat によって定義されているが、Filter の機能を拡張し、子 Filter(childFilter) として他の Filter を入れ子にして持つことができるようにする。これによって、Filter を様々な組み合わせで、より高度な Filter を実現しやすくなる。また、作成済みの Filter 同士で作成された Filter の場合、新たにライフログ DB からデータを取得し直さなくてよいという利点もある。

具体例として、2013年11月22日に、温度センサによって取得されたデータのうち、熊本県の緯度が南端から北端、経度が東端から西端の範囲内で取得されたデータを選別するための Filter は以下ようになる。

```
f0: {
  query: { date: "2013-11-22",
           device: "temperature sensor" },
  childFilter: {
    query: { s_lat: 32.09472, e_lat: 33.1953,
             s_lng: 129.9403, e_lng: 131.3294 },
    childFilter: null
  }
}
```

この childFilter に、新たな Filter を追加したり、他の Filter からこの Filter(f0) を子にしたりすることで、既存の Filter から様々な角度でデータの選別を行える。

### 4.4 A3: DisplayFormat の拡張

続いて、R3 を達成するために DisplayFormat を拡張する。DisplayFormat は Filter で選別されたデータ系列の地図上での表示形式を定義している。

これを図3の中央に示すように拡張する。前述の通り、既存手法では1つの DataSource においては、それに紐づく Filter によって選別された全てのデータに対して同様の表示形式を適用し地図上へ可視化する。そこで DisplayFormat の機能を拡張し、Filter によって選別された全てのデータについて、条件とそれが満たされた時に適用する表示形式をできるようにする。これによって、温度が一定値内の時にのみ特定の色のアイコンで地図上に可視化したり、移動中の速度によって移動軌跡の線の色を変えたりといったことが単一の DisplayFormat で可能になる。

例えば、4.3 節で利用した温度センサのデータ系列に対して、気温 27 度以上の地点には赤色、15 度以上 27 度未満の地点では緑色、15 度未満の地点では青色のマーカ (ピンのようなアイコン) をそれぞれ表示するための DisplayFormat は以下のようになる。

```
df0: {
  displayType: "MARKER", evalMode: "FIRST_ONE",
  formatRules: [
    { condition: { rank: 0,
      field: "content.temperature", exp: ">=27" },
      option: {color: "red"}
    },
    { condition: { rank: 1,
      field: "content.temperature", exp: ">=15" },
      option: { color: "green" }
    },
    { condition: { rank: 2,
      field: null, exp: true },
      option: { color: "blue" }
    }
  ]
}
```

上記のように、DisplayFormat は formatRule として、ルールを持つ。選別されたデータそれぞれに対して condition に基づき条件判定を行っていき、条件が満たされていた場合には対応する option を適用する。なお、上記の例の最下部の condition は、常に成立する条件を表している。

また、この条件の判定は condition の持つ rank 順で行う。DisplayFormat の持つ evalMode では、ルールの適用の仕方を指定しており、FIRST\_ONE と SEQUENTIAL がある。前者の場合は最初に条件を満たしたルールのみを適用、また後者の場合には条件を満たす全てのルールを順次適用していく。

これにより、各データに対して細かく表示形式を指定することができるようになる。

#### 4.5 実装

我々は、これらの拡張機能を Java 言語で実装した。プログラムの有効行数は合計 2388 行であった。データベースには MongoDB を利用し、OR マッパーとして Morphia を利用した。API 実装には JAX-RS (jersey: JAX-RS (JRS311) Reference Implementation for building RESTful Web service) を利用し、RESTful な API を提供する。

## 5. ケーススタディ

ケーススタディとして、提案フレームワークを用いてセンシングデータの可視化を行った。

### 5.1 移動中の速度センサログの可視化

拡張した MashMap フレームワークを利用して MashMap オブジェクトを作成し、可視化を行った結果を図 4 に示す。

この MashMap は、2013 年 9 月 8 日にユーザ okushi が旅行

中に速度センサにより取得したセンシングデータを線で可視化している。この線はセンサの取得した速度の値によって色が違っており、速度 10km/h 以下の区間では青、10~30km/h では緑、30~50km/h ではオレンジ、50km/h より大きい場合には赤という 4 色に色分けされて表示される。これにより、徒歩移動の部分や車移動の部分が、地図上から容易に推測できるようになった。

既存手法でこれを実現するには、色が変わる区間毎に Filter と DisplayFormat を用意し、それらを組にした DataSource を作成しなければならず、実質不可能なことであったが、機能拡張を行ったことにより、単一の DataSource で表現することができている。

この DataSource の持つ Filter(f1) 及び DisplayFormat(df1) は以下のようになる。

```
f1: {
  query: { date: "2013-09-08" },
  childFilter: {
    query: { user: "okushi", application: "sensorbox" },
    childFilter: null
  }
}
df1: {
  type: "LINE", evalMode: "SEQUENTIAL",
  formatRules: [
    { condition: c0,
      option: {strokeWeight: 3, strokeColor: "blue"}},
    { condition: c1,
      option: {strokeColor: "green"}},
    { condition: c2,
      option: {strokeColor: "orange"}},
    { condition: c3,
      option: {strokeColor: "red"}
    }
  ]
}
```

さらに、formatRules に含まれる c0~c1 の condition は以下のようになっている。ここで c0 は、常に成立するデフォルトルールになっている。

```
c0: {rank: 0, field: null, exp: true}
c1: {rank: 1, field: "content.velocity", exp: ">10"}
c2: {rank: 2, field: "content.velocity", exp: ">30"}
c3: {rank: 3, field: "content.velocity", exp: ">50"}
```

### 5.2 考察

モバイルセンシングによるデータを LLCMDM に取り込むことによって、MashMap フレームワークの仕組みにしたがって迅速に可視化が行えた。これは要件 R1 を満たす。

また、Filter の拡張により、クエリを様々な組み替えて使うことができるので、柔軟にデータの選別方法を指定できるようになった。例えば、別の日程の同様のデータソースを作成することがわかっている場合には、5.1 節で利用した f1 のように日

## Velocity Log by mobile sensing on 2013-09-08



図4 移動中の速度センサログ

付のクエリのみをもつ Filter と、それ以外のクエリを持つ Filter を別に定義しておき、必要な時に別の日付のクエリをもつ Filter を作成し、その子 Filter に既存の Filter を指定するといった使い方もできる。したがって要件 R2 を満たしたと考える。

さらに、図4を見てわかるように、一つのデータソースに対し、複数の表示形式を割り当て、それが反映された結果、道が細かく曲がっている部分や交差点などでは速度が落ちており、海際のゆるやかな道や直線的な道ではスピードを出して走行しているといったことが地図からひと目でわかるようになっている。これによって要件 R3 にも対応できたと考えられる。

以上より、モバイルセンシングデータの可視化は提案フレームワーク上で行うことが可能である。また、これにより同フレームワーク上で、同一のユーザから得られたモバイルセンシングデータ及びライフログをマッシュアップすることができるため、どのような環境下でそのライフログが取られたのかなど、より高度な行動振り返りを実現することができ、さらなる付加価値をもたらすことも可能であると考えられる。

### 5.3 限界と課題

一方で、新たな課題もある。DisplayFormat によってデータの値により細かな表示形式の指定が可能になった一方、利用オブジェクトが増えたことで可視化までの手順が増えてしまっている。これに対しては、利用方法を系統化するなどして、より一般的な表現方法についてはプリセットのようなものや、一括で作成を行える API を整備する他、XML や JSON といったテキストから各種オブジェクトを一括で生成できる機能などが挙げられる。

また、提案フレームワークを利用するにあたり、地図上に可視化を行う性質上、同じ場所で連続的に取られたデータは可視化が難しく、地図上で時間軸を表現していくような機能が必要だと考える。しかし現状においても、同様のデータが多く地点にある場合は、データを時系列で選別するのではなく、「7時

～10時」といった時間区切りでデータソースを作成するといった工夫を行えば実現可能である。

## 6. おわりに

本稿では、位置付きライフログを地図上に可視化するための MashMap フレームワークを拡張し、モバイルセンシングによって得られる様々なデータも地図上へ可視化できる枠組みを開発した。また、ケーススタディとして移動中の速度センサログを可視化し、柔軟なデータの選別や、センサの値に応じた地図上での表示方法が実現されたことを確認した。

今後の課題としては、より簡単に、より単純な手順によって様々な表現方法で可視化を行えるようにすることが挙げられる。また、MashMap フレームワークを強化していくことも引き続き必要であると考えられる。例えば、データの集計・分析を支援する機能や、位置付きライフログと、位置付きでないライフログをマッシュアップする機能、同一地図上で時系列を表現する機能を実現することが考えられる。

**謝辞** この研究の一部は、科学技術研究費（基盤研究 C 24500079, 基盤研究 B 23300009）、および、積水ハウスの研究助成を受けて行われている。

## 文 献

- [1] J.A. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M.B. Srivastava, "Participatory sensing," Proc. World Sensor Web Workshop, pp.1-5, 2006.
- [2] A.T. Campbell, S.B. Eisenman, N.D. Lane, E. Miluzzo, R.A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, and G.-S. Ahn, "The rise of people-centric sensing," Internet Computing, IEEE, vol.12, no.4, pp.12-21, 2008.
- [3] M. Ito, J. Nakazawa, and H. Tokuda, "mpath: an interactive visualization framework for behavior history," Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on, vol.1IEEE, pp.247-252 2005.
- [4] 高橋昂平, 栢本真佑, 中村匡秀, "ライフログ可視化フレームワーク mashmap の実装と評価," 信学技報, vol.112, no.305, pp.21-26, Nov. 2012.
- [5] K. Takahashi, A. Shimojo, S. Matsumoto, and M. Nakamura, "Mashmap: Application framework for map-based visualization of lifelog with location," Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT2012), pp.1-6, Nov. 2012.
- [6] 栢本真佑, 下條彰, 鎌田早織, 中村匡秀, "異種ライフログ統合のための標準データモデルとマッシュアップ api," 電子情報通信学会論文誌, vol.J95-D, no.4, pp.758-768, April 2012.
- [7] A. Shimojo, S. Matsumoto, and M. Nakamura, "Implementing and evaluating life-log mashup platform using rdb and web services," The 13th International Conference on Information Integration and Web-based Applications & Services (iiWAS2011), pp.503-506, Dec. 2011.
- [8] 齊藤裕樹, 菅生啓示, 間博人, "sbike: 参加型センシングを志向したモバイルセンシングによる自転車走行状態収集・共有機構," 情報処理学会論文誌, vol.53, no.2, pp.770-782, 2012.
- [9] ウェザーリポート Ch. <http://weathernews.jp>.
- [10] みんなでつくる放射線量マップ. <http://minnade-map.net/>.
- [11] Twitter. <http://twitter.com/>.
- [12] Flickr. <http://www.flickr.com/>.