

地図 API とタイムライン API のマッシュアップによる 時系列地理位置情報表示システムの試作

桑原 一聖[†] 服部 哲[†] 速水治夫[†]
[†] 神奈川工科大学 情報学部 情報メディア学科

近年, Google マップ API やタイムライン API などの登場により, それらの Web サービスを用いて情報を管理・表示する研究は数多く存在するようになった. しかし, それらの情報を同時に表示し, その 2 つの関係を感覚的に理解できるようなシステムの研究は少ない. 本論文では, 地理位置情報と時系列情報を表示し, その関係が感覚的に理解できるように, 地図 API とタイムライン API のマッシュアップによる時系列地理位置情報表示システムについて述べる. 評価実験の結果, 本システムを用いて時系列情報と地理位置情報の 2 つの情報を同期させて表示することにより, 各 API 単体で動作するシステムより 2 つの情報の関係が感覚的に理解できるという結果を得た.

Development of Time Series Geolocation Information Display System by Map API and Timeline API

Kazukiyo Kuwahara[†] Akira Hattori[†] Haruo Hayami[†]
[†] Faculty of Information Technology, Kanagawa Institute of Technology

Recently, a number of Web service API such as Google Maps API have been developed. And there are many systems which use these APIs. However, as long as we know, there is no system which uses both APIs to manipulate a map and a timeline. Using them together enables us to understand geolocation information along with a timeline. In this study, we propose a system using Google Maps API and TimeLine API and developed a prototype system. As the results of the experiment using our prototype, we found that handling a map and a timeline synchronously was effective for providing information.

1. はじめに

近年, インターネットによる地図サービスは珍しいものではなく, 数多くの地図サービスが存在している. 中でも Google マップはマウスのドラッグで自由に位置を変えることや, 衛星写真での地図の表示など様々な機能がある. さらに, API (Application Program Interface) の公開により, 誰でも Google マップを Web コンテンツに埋め込むことやカスタマイズすることができるようになっている.

この地図サービスを使用し, 地理位置情報を含んだ情報を管理するシステムは数多く存在するが, ²⁾ 地理位置情報以外の指標となる時系列情報を同時に表示させるようなシステムの研究はない. 時系列情報と地理位置情報の 2 つを組み合わせることは, 双方から見ても価値がある.

本論文では, 時系列情報を扱うタイムライン API と地理位置情報を扱う地図 API のマッシュアップにより, その 2 つの情報をより感覚的にわかりやす

く表示することができるインターフェースと、その際に必要になる座標情報などの入力の手簡易化ができる時系列地理位置情報表示システムの提案と試作結果を述べる。

以下、2章で研究対象の現状、3章で問題点と解決策、4章で時系列地理位置情報表示システムの試作、5章で評価実験、6章でその考察を行い、7章で結びを述べる。

2. 研究対象の現状

2.1 地図 API について

地図 API は、インターネット上にある地図サービスのプログラムを公開し、誰でも自分の Web コンテンツに地図を組み込めるサービスのことである。この代表例として Google マップ API が挙げられる。これは、Javascript を利用した Ajax (Asynchronous JavaScript + XML) が基盤となっていて、地図上をドラッグして地図を移動させることができるなどの動的な動きが可能である。単に地図を置くだけではなく、API を活用することにより地図のアイコンを変更や、地図上に線を引くことができるようになるなど、様々な機能がある。その豊富な機能により地理位置情報を用いたシステムで数多く使用されている API である。さらに最近では、Yahoo マップ API など登場してきている、基盤となっているプログラムは違うがこれも豊富な機能により様々な応用ができる。

2.2 タイムライン API について

タイムライン API は、時系列情報を時間軸上に表示させるようなインターフェースが設置できるサービスのことである。この代表例に Simile Timeline API と nifty Timeline API が挙げられる。Simile とは MIT (マサチューセッツ工科大学) のオープンソースプロジェクトのことで Javascript を利用した Ajax が基盤となっていて年表を横にスクロールするような感覚で使うことができるインターフェースで

ある。この API を活用することにより、時間軸を複数表示するなどのカスタマイズができる。nifty Timeline API は、Simile Timeline API と同じような感覚で使うことができるが、Flash が基盤であるため前者ほどのカスタマイズはできない。このタイムライン API は時系列情報を扱う研究に数多く使用されている。

2.3 関連研究

RDF/OWL を利用した歴史情報閲覧システムは、高等学校の日本史を対象に、史実と視覚的な情報をメタデータを用いて結びつけ、史実同士の関連性を視覚的に提示して歴史への理解を助ける学習支援を行うシステムである。¹⁾

このシステムは地図 API とタイムライン API を用いて高等学校での日本史での地理位置情報と時系列情報の 2 つを独立に管理・表示している。面積の変化や旅の経路などの情報を、アニメーションを用いて表示するなどの機能がある。

3. 問題点と解決策

3.1 研究対象の問題点と解決策

地図 API の問題点の 1 つとして、地理位置情報以外の指標となるデータが扱いづらいという点がある。地図 API の情報ウィンドウに時系列情報を表示することは可能だが、その情報ウィンドウは 1 つずつしか表示できないため、他の位置の時系列情報がわからなくなり時系列での把握は難しくなる。もう 1 つはデータ入力の際にある程度の専門知識が必要になる点がある。これは本論文で提案するシステム (以下、本システム) で使用する Google マップ API を例に挙げると、地図上にアイコンを表示する場合 XML (Extensible Markup Language) でデータを入力して、その XML ファイルを Google マップ API で読み込むことになる。その際、座標などのデータが必要になるのだが、XML のデータ構造に沿った形式でデータを入力

することになる点や、アイコンを表示する地点の座標を調べる必要があるためである。

タイムライン API の問題点は、時系列情報しかインターフェース上に表示ができないため他の情報、例えば地理位置情報などの把握が難しい点がある。

本システムでは、より複雑な情報を簡単に入力することができ、さらに感覚的に理解できるようにするために、地図 API とタイムライン API の 2 つをマッシュアップして開発した。それにより、互いの API 単体では表示できなかった情報を表示する。

3. 2 関連研究との関係

先行研究の歴史情報閲覧システムは地図 API とタイムライン API を使って地理位置情報と時系列情報の 2 つを扱ってはいるが、その情報を同時に表示しておらず、データの入力に専門的な知識が必要である。これに対し、本システムでは、地図 API とタイムライン API を同期的に動作し、さらに専門的な知識が必要なくデータ入力ができる。

4. 時系列地理位置情報表示システムの試作

4. 1 システム概要

本システムは、地図 API である Google マップ API とタイムライン API である Simile Timeline API のマッシュアップにより 2 つの API を同期させ、時系列情報と地理位置情報を同時に表示することができるシステムである。

大きく分けてデータ管理部とデータ表示部 2 つのモジュールに分けることができる

システムの具体的な流れはまずシステムの管理者がデータ管理部から座標入力の簡易化をされたデータフォームで時系列情報と地理位置情報、その 2 つに付随した情報を入力する。その情報はデータベースに登録され、データ管理部でデータリストと削除ボタンが追加されるようになる。データ表示部では、データベースの情報を各 API で表

示できるように物語毎に分けて XML に動的に生成される。その XML を読み込むことで各 API にデータ表示され、同期的に動作する。

以下にシステム構成図を示す。

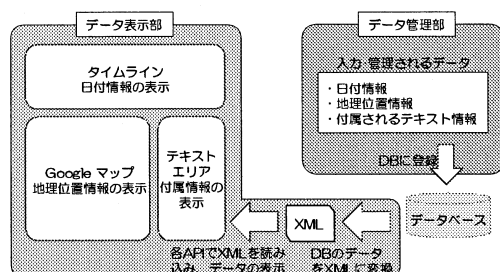


図4. 1 システム構成図

4. 2 データ管理部概要

データ管理部では、地理位置情報と時系列情報の入力をするデータ入力部と、キーワードから座標を検索できる座標検索部、データのリストと削除ボタンが表示されるデータリスト部に分けられる。

以下は、データ管理部の全体画面を示す。

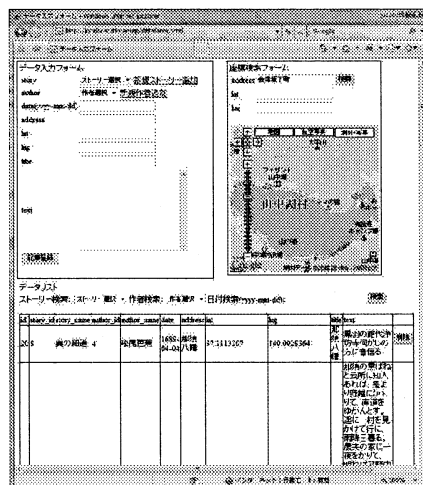


図4. 2 データ管理部画面

4. 2. 1 座標検索部

座標検索部は地理位置情報に必要な座標

入力簡易化をするためのものである。使用の流れは、まず登録したい場所を場所欄に記入して検索する。検索するとその検索された場所が Google マップの中央に表示され、同時に緯度・経度が表示される。その際座標検索部の場所・緯度・経度欄だけではなく、データ入力部の場所・緯度・経度欄にも同じデータが自動で入力される。検索された場所よりも正確な位置を入力する場合は Google マップ上を直接クリックすると、そのクリックされた位置の緯度・経度が表示される。

場所検索には Google マップ API に搭載されているジオコーディング機能を使用している。ジオコーディングとは、ある地名や住所から緯度・経度が検索される機能のことである。

4.3 データ表示部概要

データ表示部では、データ管理部でデータベースに登録されたデータを XML の形式に変換し、その XML を地図 API とタイムライン API の両方で読み込みデータを表示させる。その際地図 API とタイムライン API は同期的に動作するようになっている。

以下は、データ表示部の全体画面を示す。

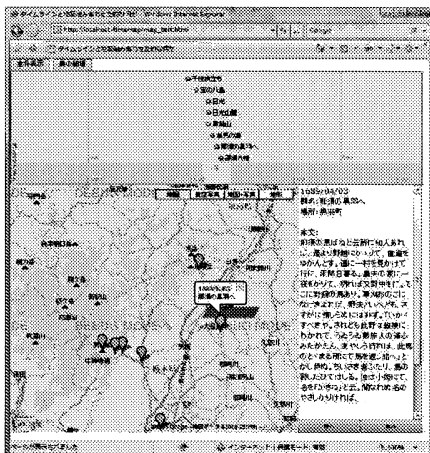


図4.3 データ表示部画面

4.3.1 データベースの内容をXMLに変換

地図 API とタイムライン API はデータの読み込みの際 XML の形式でデータ読み込みを行う必要がある。そのためデータベースのデータを、PHP を用いて XML の形式に自動生成するようなプログラムを作成した。ストーリー毎のデータの表示のために、現在登録されているストーリーの数だけ画面上部にボタンを設置し、そのボタンをクリックすることによって、そのストーリーの記事が地図 API とタイムライン API の両方に表示されるようになる。

以下は、PHP で生成される XML のデータ構造を示す。

```
<?xml version="1.0" encoding="UTF-8" ?>
<result>
  <item>
    <itemid>アイテム ID</itemid>
    <pubDate>日付</pubDate>
    <address>場所</address>
    <lat>緯度</lat>
    <lng>経度</lng>
    <title>題名</title>
    <text>本文</text>
    <story>物語名</story>
    <author>作者名</author>
  </item>
  . . .
</result>
```

4.3.2 各 API の同期

地理位置情報と時系列情報の 2 つを感覚的に理解できるようにするためには、地図 API とタイムライン API が同じ情報を表示している必要がある。そのために地図 API で表示されているデータアイコンをクリックすると、そのデータが地図上の真ん中に表示されると同時にタイムライン API でも同様のデータがタイムライン上の中央に表示されるよう

になる。タイムライン上のデータアイコンをクリックしても上記と同様にクリックされた情報がタイムライン上の中央に移動し、地図 API でも同様のデータが地図上に表示される。その際、地図 API は Flex を用いた Flash アプリケーション、タイムライン API は Javascript を用いて記述されているため、Flash と javascript 間での変数の受け渡しが必要になる。

変数の受け渡しが行われるのは以下の条件のときである。

- ・地図上のデータをクリックする。
- ・タイムライン上のデータをクリックする。
- ・記事送り・戻しボタンをクリックする。
- ・ストーリーを読み込む。

この地図 API とタイムライン API の同期化により、地理位置情報と時系列情報の 2 つを感覚的に理解できるようになっている。

5. 評価実験

5.1 評価方法

本システムは、データ管理部、データ表示部という 2 つの大きなモジュールがあるため、データ管理者とデータ閲覧者両方の視点から評価実験をする。評価方法はデータ管理部、データ表示部共にアンケート調査を行う

5.1.1 データ管理部の評価

データ管理部の評価は、予め用意したデータを被験者にデータ入力をしてもらった。その後、データ入力部・座標検索部・データリスト部の 3 つに分け、操作性の段階評価、上記項目以外で気になる点を記述するアンケート調査を行う。

5.1.2 データ表示部の評価

データ表示部の評価では、予め、地図 API 単体で動作するシステムとタイムライン API 単体で動作するシステムの 2 つを作成し、本システムと合計し

た 3 つのシステムにモデルデータとして松尾芭蕉の奥の細道の情報を入力したものを用意する。被験者は、その 3 つのシステムを閲覧し、システムごとの比較を行う。その後、各システムで時系列情報と地理位置情報の関係の理解度と本システムの操作性を段階評価、上記項目以外で気になる点を記述するアンケート調査を行う。

5.2 評価結果

学生 10 人に、本システムを操作してもらい意見・評価を得た。

5.2.1 データ管理部

データ管理部を 3 つの部に分け、それぞれの操作性を 5 段階で評価した結果の平均点は、データ入力部で 3.7、座標検索部で 4.1、データリスト部で 3.2 という結果となった。

5.2.2 データ表示部

本システムと各 API が単体で動作するシステムの比較し、時系列情報と地理位置情報の関係が感覚的に理解できるかを評価した結果は、本システムが 4.6、地図 API 単体でのシステムが 3、タイムライン API 単体でのシステムが 2.7 となり、本システムの操作性は 4.2 という結果だった。

6. 考察

データ管理部での評価で、座標検索部の操作性が高い評価だったという結果から、本稿の目的の 1 つである、座標情報の入力の簡易化は効果的に行われていたと言える。

データ表示部の評価としては、地図 API 単体のシステムやタイムライン API 単体のシステムより、本システムの方が時系列情報と地理位置情報の関係を感覚的に理解できるという評価となった。この結果から本稿の目的である、時系列情報と地理位置情報の関係が感覚的に理解することがで

きるシステムを試作することができたと言える。

しかし、自由記述のアンケートでタイムラインの間隔が調節できないため時系列情報の把握が難しい場合があるとの意見が挙げられたように、記事の時間間隔が大きい場合タイムライン上に記事が 1 つしか表示されないという問題点も見つかった。その問題を解決するためには物語毎に記事の時間間隔を測定し動的にタイムラインの間隔を計算し、反映するような機能が必要になると考えられる。

本システムは、地理位置上の変化を伴う、時系列情報ならば本システムで表示可能のため、今回モデルデータとした奥の細道のように歴史の教材としての使用方法だけではなく、小説や自伝での利用や台風情報などの天候情報での利用など、様々な応用方法があると考えられる。

今後は、上記の問題点の解決とセキュリティの強化により、実用化ができるように改善していき、さらに隣接する記事を地図上で線を繋ぐなどの機能を追加することにより、より様々な使用方法ができるよう開発を進める。

7. おわりに

地図 API とタイムライン API のマッシュアップによる時系列地理位置情報表示システムの提案と試作を行うことにより、時系列情報と地理位置情報の 2 つの情報の関係を感覚的に理解することが可能となった。

この結果から時系列情報や、地理位置情報だけに留まらず、1 つの情報を表示するシステムを複数組み合わせることにより、より多くの情報を表示し、理解することができるシステムが開発できる可能性があると言える。

参考文献

- 1) 馬場裕子, 永森光晴, 杉本重雄: RDF/OWL を利用した歴史情報閲覧システムの構築, 情

報処理学会研究報告. 人文科学とコンピュータ研究会報告, Vol2008, No.73, pp.49-56.

- 2) 服部哲, 後藤昌人, 安田孝美, 横井茂樹: 地図サービスと CMS の連携による安全情報共有システム, 情報処理学会研究会報告. グループウェアとネットワークサービス研究会報告, Vol2007, No. 32, pp. 43-48.

- 3) Google マップ:
<http://www.google.com/apis/maps/>

- 4) Simile Timeline API:
<http://simile.mit.edu/timeline/>