

位置適応サービスのパーソナライズを容易に実現する 行動履歴解析システムの設計と実装

伊藤 昌毅¹ 徳田 英幸^{1,2}

¹ 慶應義塾大学大学院 政策・メディア研究科 ² 慶應義塾大学 環境情報学部

ユビキタスコンピューティング環境では、日常生活のさまざまな活動が環境に備えられたセンサや携帯するデバイスによって取得され、こうした情報を活用することでユーザの行動に適応したさまざまなシステムの開発が実現する。本稿では、蓄積された多様な行動履歴を柔軟に解析する行動履歴解析システムを構築する。本システムでは入力された行動履歴の解析方法をビジュアルプログラミングの手法で設計するため、アプリケーションごと異なるさまざまな解釈手法を容易に構築できる。本稿では行動履歴解析システムの設計と実装について述べ、また本システムを用いた際の性能を評価する。

1. はじめに

環境に計算機能やセンサが埋め込まれ人間活動を支援するユビキタスコンピューティング環境 [1] では、日常生活のさまざまな行動情報の自動取得や記録が容易になり、記録された情報を活用することで新しいアプリケーションの可能性が広がる [2]。現在、日常から持ち歩く携帯電話に計算機能やカメラ機能、GPS を利用した位置取得機能が追加され、また FeliCa を用いた鉄道の乗車機能の追加も検討されることで、こうした環境の実現が現実的になっている。行動履歴を解析しその結果を用いることで、ロケーションアウェアネスを実現するシステム [3] において単に位置だけでなく、ユーザのより詳細な行動への適応が実現する。本稿では、こうしたシステムを柔軟に構築する行動履歴解析システムを提案し、提案したシステムの設計、実装および評価について述べる。

本稿の構成は以下のとおりである。2 章において行動履歴を解析しアプリケーションで活用する際の問題点を整理する。3 章で柔軟な行動履歴解析システムを提案し、その機能や利用方法を説明する。そして、4 章においてシステムを用いて構築したアプリケーションを紹介する。5 章で設計と実装を述べた後、6 章で本システムのパフォーマンスを評価する。7 章で関連研究を紹介し、最後に 8 章において本稿をまとめる。

2. 行動履歴の解析と活用

本稿では、位置情報、時刻情報、および行動内容の 3 つの要素を備えた情報を行動情報と呼び、過去に取得された単体あるいは複数からなる行動情報の集合を行動履歴と呼ぶ。例えば GPS など得た移動軌跡や、場所と時刻とを伴った買い物情報などがこうした行動履歴の一例である。

さまざまなセンサから取得した情報を行動履歴としてまとめ、行動履歴を解析してユーザの行動や嗜好を把握

することで、ユーザに過去の行動を視覚的に提示したり、さまざまなシステムをユーザに適応的に振舞わせることができる。以下に商店街におけるアプリケーションの例を用いて具体的に説明する。そして、こうしたシステムを構築する際に必要な機能を論じる。

2.1 アプリケーション例

商店街において複数の店を回遊しながら買い物をする場面で、街頭の情報端末がユーザに適した情報を提示するシステムをアプリケーション例に挙げる。図 1 にシステムの全体像を示す。街頭端末は、ユーザの携帯端末や環境に埋め込まれたセンサから、ユーザのさまざまな行動情報を取得し、ユーザに適切な情報提示を行う。



図 1: 街頭情報端末を用いたアプリケーション例

行動履歴の表示

これまでに買い物をした店や、買い物の途中で手に取った品物、複数の品物の比較など、買い物をしている間の行動をさまざまな観点でユーザに提示することで、ユーザの品定めを支援する。整理された情報を提示することで、買い忘れや買いすぎを注意したり、同行者がいる場合に複数人での情報の共有を容易にする。

行動履歴に基づいたアプリケーションの動的適応

すでに済ませた買い物や今まで回った店、長時間滞在した店といった行動履歴からユーザの趣味や買い物の目的を推測し、ユーザに商店や商品を推薦する。情報端末

An Action History Mining System for Personalization of Location-aware Services

Masaki Ito¹, Hideyuki Tokuda^{1,2}

¹ Graduate School of Media and Governance, Keio University

² Faculty of Environmental Information, Keio University

5322, Endo, Fujisawa, Kanagawa 252-8520, Japan

E-Mail: <niya@ht.sfc.keio.ac.jp>

や携帯デバイスに広告を表示したり、情報端末同士が連携しナビゲーションを行うなど趣味や目的に適応したアプリケーションによりユーザの買い物を支援する。

2.2 行動履歴解析システムの機能

例に挙げたように、行動履歴を解析してこれまでの行動を表示したりアプリケーションを適応的に動作させるためには、以下のような機能が必要となる。

多様な行動履歴への対応

GPS などの単一のデバイスから取得した移動軌跡だけでなく、さまざまなデバイスやインフラから取得された多様な行動履歴を、同一のシステムで取り扱う必要がある。例えば、Suica のような鉄道乗車用の IC カードに記録された乗車記録や、電子的に支払われた買い物記録など、さまざまな形態の行動履歴が考えられる。また時刻を移動軌跡と比較することで、デジタルカメラによる写真撮影といった時刻情報のみを持つ情報を行動情報として扱えるようになる。このような、多様な行動履歴を総合して取り扱う仕組みが必要である。

また、取得した行動履歴を解析するために、行動情報と並んでさまざまな地理情報を同時に取り扱う必要がある。センサから得られた行動情報に加えて、鉄道やバスの路線図や、建物の形状や店名、道路などが記述された地理情報を用いることで、行動履歴の高度な解釈も可能になる。

行動履歴を取得し解析する既存の研究は、Active Badge[4] を利用した研究 [5] や無線 LAN の電波強度を利用した研究 [6]、GPS を利用した研究 [7][8][9][10] などが挙げられる。しかし、いずれの研究においても単一のデバイスのみを利用を想定している。また、位置情報デバイスから取得した情報を解釈するための地理情報としても、限られた観光地の情報 [7] やバス路線の情報 [10] など、特定のアプリケーションに特化した地理情報が利用されている。

多様な解析手法の実現

多様な観点から行動履歴を視覚化したり、さまざまなアプリケーションを行動履歴の解析結果に基づき適応的に動作させるためには、複数の行動履歴の解析手法が必要になる。たとえばあるアプリケーションにおいては典型的な行動パターンを必要とし、別のアプリケーションでは特定の場所に訪れた回数を必要とするだろう。こうした要求に対応するためには、さまざまな解析手法を柔軟に開発できる仕組みが必要である。

観光地の訪問パターンの抽出 [7] や交通分析に適した形式による行動の抽出 [8]、生活を補助するシステムのための日常生活パターンの抽出 [10] など、既存の研究においてもさまざまなアプリケーションが提案されそのための解析手法が議論されている。しかし、これらの研究は特定のアプリケーションのための解析手法の洗練に主眼が置かれており、多様なアプリケーションを想定していない。

3. 柔軟な行動履歴解析システム

本稿では、さまざまな行動履歴を入力でき、行動履歴の解析手法を柔軟に開発できる行動履歴解析システムである mPATH システムを提案する。mPATH システムは、

マウス操作による入力行動履歴の選択や解析手法の構築を可能にするビジュアルプログラミングインタフェースを提供し、直感的な手法による行動履歴の解析を実現する。行動履歴の解析時の補助情報として、行動履歴に加え地理情報も取り扱う。本システムはまた、構築したアプリケーションを動作させるプラットフォームとしても機能し、アプリケーションの容易な開発や配布を実現する。図 2 に、本システムのスクリーンショットを示す。

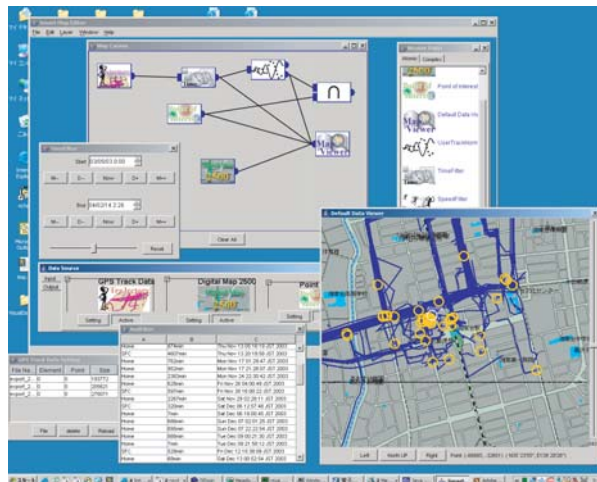


図 2: mPATH システム

3.1 mPATH システムの特徴

解析手法の部品化

行動履歴の解析はデータの分類や計量、平均化など様々な手法を組み合わせて行う。本システムでは、こうした手法を部品化し、部品の組み合わせで解析を実現する。このことにより、複雑な解析手法の開発が容易に行える。

ビジュアルプログラミングによる平易な開発

部品化された解析手法の組み合わせを行うために、データフローに基づいたビジュアルプログラミング環境を提供する。解析手法が視覚化されるため、解析手法の開発が直感的に行えるようになる。

任意の位置表現による入出力

行動履歴の種類により、位置情報の表現形式は異なる。緯度経度による座標系や、地名や施設名といった空間につけられた名称、また時刻情報しか含まず他の行動履歴と照合することで位置情報が得られる場合も考えられる。本システムでは、入力された様々な座標系による行動履歴を緯度経度による地理座標系に変換する。行動履歴の座標系を統一することで、任意の座標系による行動履歴を同一のシステムで取り扱えるようにする。

Push, Pull 両用の解析機構

アプリケーションの要求によって行われる解析を Pull 型の解析と定義し、入力されている行動履歴の変更を引き金にして行われる解析を Push 型の解析と定義する。アプリケーションの種類によって、解析の対象となるデータは事前に蓄積された行動履歴の場合もセンサなどから逐一送られてくる情報の場合もあり、それによって必要

となる解析手法も異なる。本システムは Push と Pull 両方の解析に対応することで、多様なアプリケーションの構築に対応する。

3.2 データフローに基づくビジュアルプログラミング

mPATH システムは、行動履歴解析手法の開発のためにデータフローに基づいたビジュアルプログラミング環境を提供する。図 3 に、mPATH システムの提供するプログラミング環境のスクリーンショットを示す。

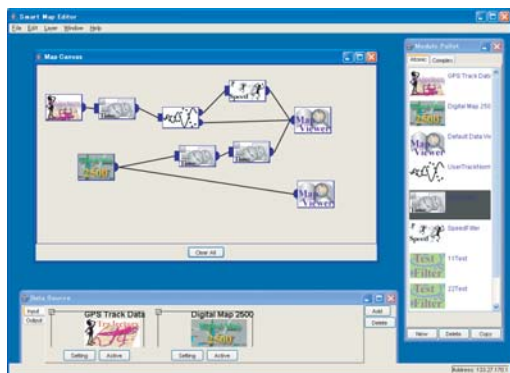


図 3: ビジュアルプログラミング環境

mPATH システムでは、行動履歴や地理情報の入力機能や入力された情報の解析手法、そして解析結果の出力機能がそれぞれアイコンとして表現されている。それぞれのアイコンはデータの入出力機能を備え、アイコン間を接続することによりデータが交換されるようになる。図 3 左上のウィンドウがプログラムのためのウィンドウであり、図 3 右側のウィンドウに登録されている入出力機能や解析機能をドラッグアンドドロップし、表示されたアイコン同士をマウスのクリックにより接続することでプログラムを構築できる。複数の解析手法を接続することで、複雑な解析が実現する。解析手法に対する変更は即座にシステムに反映され、対話的な解析を実現する。

3.3 行動履歴の入力

行動履歴や地理情報を示すアイコンを利用することで、mPATH システムへの情報の入力を実現する。ネットワークから取得する行動軌跡やファイルとして保存されているデータ、直接接続されたデバイスから入力されるデータなどさまざまなデータが考えられるが、本システムではすべてアイコンとして表現され同等に取り扱う。以下に述べるデータの入力機能が現在実装されている。

GPS 軌跡入力機能

GPS により取得されたデータを読み込む。現在はハンディ型 GPS の Garmin eTrex[11] で取得したデータファイルの読み込みに対応しているほか、シリアルポートを経由した NMEA-0183 形式の情報のリアルタイムでの読み込みに対応する。

写真入力機能

デジタルカメラで撮影したファイルから、EXIF[12] 情報として書き込まれている日時と位置に基づき写真撮影という行動情報を生成し本システムに入力する。位置情報を持たないファイルの場合、GPS 軌跡情報を併用する

ことで時刻を基に写真データの位置情報の推定を行う。この機能により、GPS を持たない通常のデジタルカメラによる写真撮影行動の取得を実現する。

数値地図 2500 入力機能

国土地理院が GIS での利用を目的に発行しているベクタ形式の地図である数値地図 2500 を読み取り、本システムへ入力する。これらのデータは、地図として視覚化されるほか、行動軌跡と建物情報などを対応させ、GPS から得られた情報からより高次元の情報を抽出する際にも用いられる。

Shape ファイル入力機能

Shape 形式 [13] は、GIS において広く利用されている米 ERSI 社の ArcGIS[14] において地図データを格納する際に用いられるファイルフォーマットである。本機能を用いることで、市販されている Shape 形式の地図データを本システムで利用できる。

3.4 行動履歴の解析

行動履歴の解析手法がアイコンとして登録されており、これらを利用することで行動履歴を選別したり、情報を抽出したりすることができる。解析手法は入出力を持つフィルタとして表現されており、複数の解析手法を組み合わせることができる。現在、以下に述べるフィルタが解析手法として登録されている。

時刻フィルタ

入力された情報を、時刻情報に基づいて選別する。GPS 軌跡データに適応して特定の期間の軌跡のみを抽出したり、地理データに適応して特定の時点での地理情報を取得するなどの用途が考えられる。

スピードフィルタ

スピード属性を持つデータに適用して、特定の範囲のスピードのデータのみを抽出する。GPS データに適用し、乗り物の推定などより高度な情報の抽出に用いることが出来る。

軌跡正規化フィルタ

軌跡データから、出発点、移動軌跡、終着点からなる構造を切り出し出力するフィルタである。とくに GPS で取得したデータは電波の受信状況の変動によるデータの欠落やキャリブレーションのためのデータ取得開始までの遅れなどがあり、こうしたデータの不備を補完しながら軌跡データを正規化する。本フィルタからは正規化された軌跡情報を出力するほか、軌跡中の一定時間以上滞在した地点のデータを点データとして出力する。

対応付けフィルタ

対応付けフィルタは二つの入力を持つ。一方に入力された点情報に対し、もう一方から地理的に対応する要素を検索し、点情報に対応する要素の情報を付与する。本フィルタに地図情報と点情報を入力することで、任意の点に対応した建物の名称や地名といった情報を付与することができる。

カウントフィルタ

本フィルタは、内部に格子状に区切られた地理座標系を持っており、入力された情報の個数を各格子ごとに数える。出力は数値情報を伴った格子情報として出力され

る．本フィルタは軌跡や点の個数を数え上げるのに用いられる．

3.5 解析結果の出力

行動履歴の解析結果は出力機能を表すアイコンに接続され、解析結果を様々なシステムに出力する．出力先のひとつとして視覚化アプリケーションを構築するほか、ファイルに出力したりネットワークを用いて他のシステムに解析結果を転送するなどの方法で、分散システムにおける解析結果の利用を実現する．

以下に、現在利用できる出力機能を挙げる．現在は視覚化による出力のみを実現しているが、今後出力機能を充実させネットワークを利用した他のシステムへの出力などにも対応する．

標準地図出力機能

データを標準的な地図と同様の2次元座標上で表示する出力機能であり、行動軌跡の地理的な側面を表現するのに有効な出力機能である．複数のデータの入力が可能で、任意の地点のデータを任意の縮尺、表示順序で表示する．図4に表示例を示す．本例では、数値地図2500とGPS軌跡のそれぞれの情報を重ねて表示している．

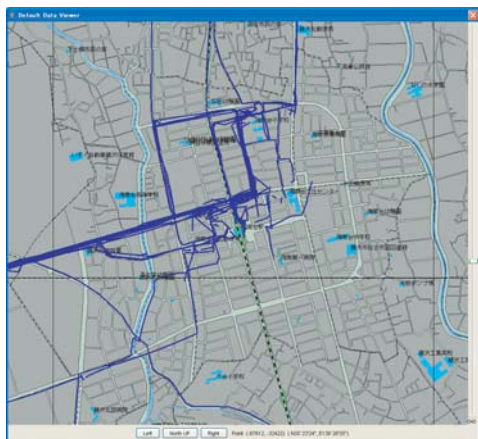


図4: 標準地図出力の例

表出力機能

データを表として出力する機能であり、データの詳細を数値や文字列として表示しながら把握できる．行動履歴解析手法の開発の際に有効な情報を得られる．図5に本機能の表示例を示す．

重み付け地図出力機能

標準地図出力機能に加え、地理的な領域ごとに設定された数値を入力し、重みとして視覚化する機能である．重みの表現手法には色や濃度による表現や、多次元尺度構成法などを利用した歪みをもった地図などの表現が考えられる．現在の実装では、重みが設定されている区域ごとの縮尺として表現している．図6は、本手法を後述するユーザの行動による重み付けアプリケーションで用いた例である．

Place	Time	Start
Home	10 hours 37 min	Thu Jan 14 10:00:00 JST 2004
SFC	10 hours 40 min	Thu Jan 15 11:18:50 JST 2004
Alome	55min	Fri Jan 16 22:36:34 JST 2004
Home	16 hours 43 min	Sat Jan 17 21:52:29 JST 2004
Alome	1 hours 7 min	Sun Jan 18 21:28:26 JST 2004
Home	11 hours 13 min	Sun Jan 18 22:42:20 JST 2004
Home	55min	Mon Jan 19 20:00:01 JST 2004
Home	13 hours 9 min	Mon Jan 19 22:07:14 JST 2004
Home	17 hours 19 min	Tue Jan 20 20:20:03 JST 2004
Home	16 hours 18 min	Wed Jan 21 23:23:16 JST 2004
SFC	5 hours 25 min	Thu Jan 22 16:13:20 JST 2004
Home	15 hours 30 min	Fri Jan 23 01:37:36 JST 2004
Home	10 hours 19 min	Sat Jan 24 03:52:16 JST 2004
Alome	55min	Sat Jan 24 22:14:51 JST 2004
Home	22 hours 2 min	Sat Jan 24 23:26:52 JST 2004

図5: 表出力の例

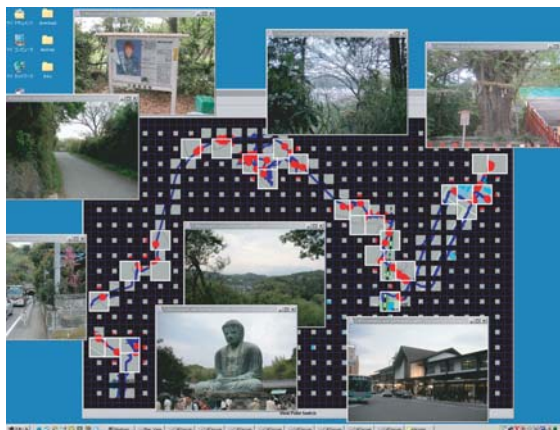


図6: 重み付け地図出力の例

4. アプリケーションの構築

mPATHシステムを用いて、ユーザの行動履歴に基づいてパーソナライズされた情報提示を行うアプリケーションを構築した．

4.1 地図を用いた行動軌跡の表示

GPSから得られた軌跡から特定の期間の軌跡を選別し、閾値以上の時間滞在した場所を点として抽出、地図と重ねて出力するアプリケーションを構築した．解析部品の接続状態を変更することで、地図上に線として表現される軌跡情報を表示するなど表示情報の変更ができる．図7にプログラムを示す．入力したGPSデータに時刻フィルタおよび軌跡正規化フィルタを適用したのち、数値地図2500とともに標準地図出力機能を用いて出力している．

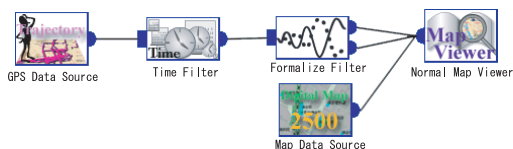


図7: 地図を用いた行動軌跡表示のプログラム

4.2 滞在地点のリスト

ユーザが長時間滞在した地点を抽出し、地点の名前とともに表形式で出力するアプリケーションを構築した．GPS軌跡に対し軌跡正規化フィルタおよび対応付けフィ

ルタを適用し、出力には表出力機能を用いた。図 8 にプログラムを示す。図 5 は本アプリケーションの出力例である。長時間滞在した地点は、ユーザが何らかの行動を起こした場所であったり、行動の出発点もしくは終点であると考えられ、本例でも自宅や大学などが抽出されている。

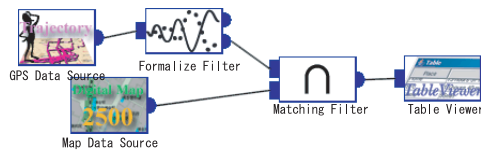


図 8: 滞在地点リストのプログラム

4.3 ユーザの行動による重み付け

ユーザの行動に基づいて空間を重み付けし、重み付けを視覚化した地図を表示するアプリケーションを構築した。本アプリケーションは観光や買い物といった行動を対象とし、ユーザのこれまでの行動を直感的に表現するアプリケーションである。本アプリケーションで用いる重み付けの手法は、行動が行われた領域を小さい矩形に区切り、それぞれの矩形ごとにユーザの通過回数や買い物や写真撮影の回数などを数え、それを合計したものを各矩形の重みとする。

図 6 は本アプリケーションを観光記録に適用した例であり、主に写真撮影に着目して重み付けを行っている。本アプリケーションはまた写真データの表示機能も備え、行動に関連付けられた写真ブラウザとして機能する。

図 9 に示したプログラムのように、本アプリケーションの構築に際し GPS 軌跡入力機能、写真入力機能、数値地図 2500 入力機能を用いて情報を入力し、軌跡正規化機能及びカウントフィルタを用いて情報を解析した。情報の出力には重み付け地図出力機能を用いている。

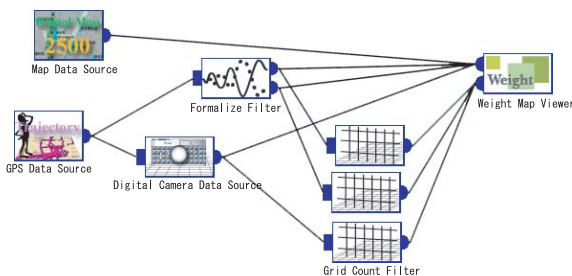


図 9: ユーザの行動による重み付け地図のプログラム

5. mPATH システムの設計と実装

mPATH システムは GUI アプリケーションとして設計され、14,000 行程度の Java 言語により実装した。mPATH システムは、図 10 に示すように情報入力部、情報出力部、行動履歴解析部、解析部品管理部、解析手法プログラム部から構成される。本節では、mPATH システム内部で利用されるデータ形式と、入出力機能や解析機能の設計について述べ、本システムの特徴であるモジュール化された解析の実現手法を説明する。

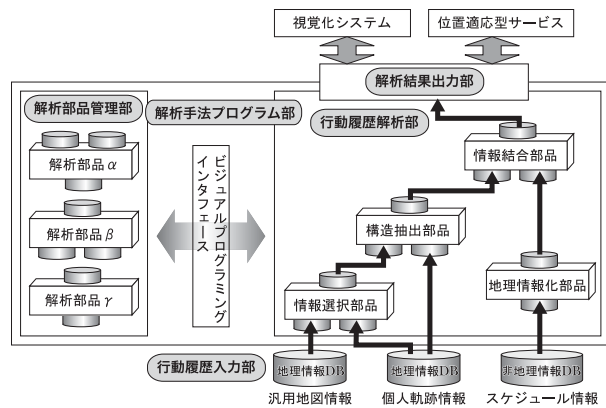


図 10: システム構成図

5.1 行動履歴のデータ形式

様々な行動履歴や地理情報を表現する共通データ形式として、本システムでは ActionElement クラスを定義する。ActionElement クラスは本システムで取り扱うすべての行動履歴情報や地理情報を表現し、本システムに入力されるデータはすべてこのクラスのサブクラスとして実装される。図 11 にクラス図を示す。ActionElement クラスは、入れ子状のデータ形式を持つことで複数の要素からなる行動履歴や地理情報を表現する。たとえば GPS から取得した軌跡情報を表現する場合、出発点から終点までの一連の軌跡が一つの ActionElement クラスとして表現される。さらに一連の軌跡を構成している通過点の一つ一つが、同一の形式で表現され InnerElements として保持される。行動履歴解析部品の開発の際には、InnerElements を辿ることで解析に必要な粒度の情報を得ることができる。また、すべてのデータが共通の型で表現されるため容易な開発が実現する。

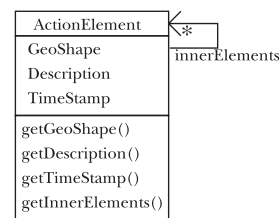


図 11: ActionElement のクラス図

5.2 部品化された解析手法の構造

本システムでは、情報入力部、情報出力部、行動履歴解析部をすべて、共通データ形式の入出力機能を任意の個数持った ActionFilter クラスとして設計、実装する。個々の情報入力、出力、解析機能は抽象クラスである ActionFilter クラスのサブクラスとして実装される。図 12 に、ActionFilter クラスと関連するクラスを示したクラス図を示す。

ActionFilter クラス

ActionFilter クラスは抽象クラスであり、ActionFilter の基本的な機能である ActionFilter 同士の接続関係を実

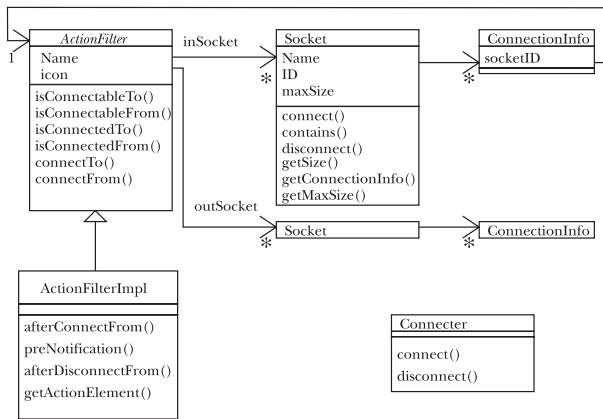


図 12: ActionFilter のクラス図

現する。ActionFilter は任意の個数の入力 Socket と出力 Socket を持ち、ActionFilter 同士の接続は入出力 Socket 同士の接続として表現される。

Pull 型の解析の際には、最下流の ActionFilter を起点に上流の ActionFilter の解析結果を返すメソッドを順次呼ぶことで履歴解析が行われる。本システムでは、すべての解析処理は内部的に Pull 型として実現されており、上流から下流へは変更の通知のみが行われる。変更通知を受けた最下流の ActionFilter が Pull 型の行動履歴解析を行うことで、仮想的に Push 型の解析処理を実現している。

Socket クラス

Socket クラスは ActionFilter の情報入出力機能を表現するクラスであり、各入出力に対応して接続可能数の上限を保持した Socket インスタンスがつけられる。ActionFilter 同士の接続は、ActionFilter の入出力 Socket を指定することで行われ、接続状態は ConnectionInfo クラスとして保持される。

Connector クラス

Connector クラスは、ActionFilter 同士の接続を実現するクラスである。Connector クラスは接続と切断のためのメソッドが用意されており、解析手法をプログラムするモジュールは入出力 ActionFilter、入出力 Socket 番号を指定することで ActionFilter 同士の接続を行う。

ActionFilterImpl クラス

ActionFilter の実装の際は以下のメソッドを実装する。

- **getActionElement(GeoShape area):** ActionFilter に解析結果をを要求するメソッドで、主に下流に接続されている ActionFilter から呼び出される。引数にはデータを必要とする地理的な領域情報が渡されるので、戻り値として渡された領域と重なるすべての解析結果を ActionElement の形で返す。
- **afterConnectFrom(ActionFilter fromFilter):** 上流側の Socket に新たに ActionFilter が接続されたときに呼び出されるメソッドであり、引数として新たに接続された ActionFilter が渡される。

- **afterDisconnectFrom(ActionFilter fromFilter):** 上流側の Socket に接続されていた ActionFilter が切断されたときに呼び出されるメソッドであり、引数として接続されていた ActionFilter が渡される。
- **preNotification(ActionFilter fromFilter):** 上流側の Socket に接続されていた ActionFilter の変更時に呼び出されるメソッドであり、変更された ActionFilter が引数として渡される。本メソッドの処理後、下流の ActionFilter に変更が通知される。

6. 評価

実装したシステムの動作時のパフォーマンスを測定し、本システムが目指す対話的な行動履歴解析手法の開発に支障のない応答速度が得られるか確認する。また、今後のシステム改良点を探すためのデータとして利用する。

測定には表 1 の環境を利用し、表 2 のデータを入力して測定した。GPS による軌跡データは、筆者が実際にハンディ GPS を持って収集した情報である。

表 1: 測定環境

CPU	Pentium4 2.53GHz
memory	1024MB
OS	Linux 2.4.22
JDK	J2SDK 1.4.2_03

表 2: 利用データ

種類	GPS 軌跡データ	数値地図 2500
説明	2003 年 6 月 ~ 11 月	藤沢市全域
サイズ	1,956,381 byte	9,261,000 byte

6.1 解析時のパフォーマンス測定

行動軌跡の解析時に、ハードコーディングされた解析アルゴリズムに対し本システムにおける部品化された解析手法を接続する手法がどの程度オーバーヘッドとなるか考察する。

軌跡情報の閲覧アプリケーションに対し、時間制約を加えるフィルタを経由させ特定の時間内の軌跡の閲覧を実現できるシステムを構築した。時刻フィルタを多段に接続し、本システムのオーバーヘッドを測定する。

ハードコーディングされたシステムにおいては、このような解析条件において、解析時間はごくわずかず直線的に増加すると考えられる。本システムにおいて解析に要する時間の変化を測定することで、本システムの解析アルゴリズム以外の部分のオーバーヘッドについて考察する。

本測定の結果を図 13 に示す。この結果に示されているように、時刻フィルタを 0 個から 15 個に増加させても、わずかな所要時間の増加しか見られない。よって、このような解析部品の多段接続における本システムオーバーヘッドはわずかなものであると考えられる。

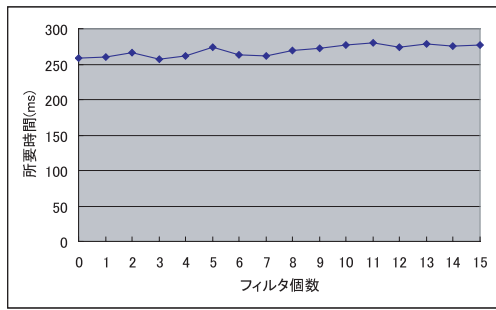


図 13: オーバーヘッドの測定

6.2 アプリケーションのパフォーマンス測定

本システムを用いて構築するアプリケーションの単純なものとして、GPS 軌跡入力モジュールと数値地図 2500 モジュール、Default Viewer を組み合わせ、軌跡および地図を重ねて閲覧するアプリケーションを構築し、レンダリングに要する時間を測定、図 14 にまとめた。なお、軌跡、地図単体でのレンダリング時間も同時に示してある。

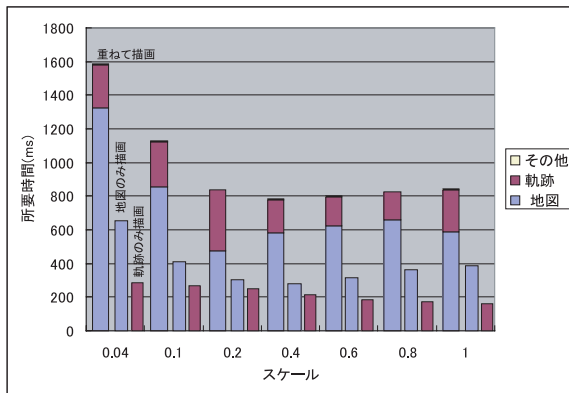


図 14: パフォーマンス測定結果

大縮尺の場合にも、描画時間は 800ms 前後かかっており、また小縮尺の場合には 1500ms 以上の描画時間を要する。地図のみ、または軌跡のみで測定した結果と比べると、特に地図の描写において地図のみの場合の 2 倍前後の時間を要しており、これが全体の描画時間を大きく損ねている。今後、小縮尺時のレンダリング手法の改良や、重ねて表示したときのパフォーマンス低下の改善を図る必要がある。

7. 関連研究

本稿において構築したシステムと直接対比できるシステムはないが、本研究が前提としている研究や、本研究と一部で問題意識が重なっている研究は多数存在する。こうした研究を紹介しながら本研究と対比する。

7.1 移動軌跡の解析によるアプリケーションの実現

既存研究として述べた PEPYS[5] や Cyberguide[7]、Activity Compass[10] は、それぞれセンサから得られた移動軌跡から日記を生成したり、観光ガイドや日常生活の補助を行うシステムである。これらの研究においては、特

定のデバイスやアプリケーションに特化して、有効な移動軌跡の解析手法を検討している。センサデータからより高次元の行動情報を抽出したり、行動予測のための行動モデルの構築などが行われている。他に文献 [9] や文献 [8] の研究においても、GPS で取得し蓄積した行動履歴を対象とし、それを解析することで行動パタンの抽出や行動予測などを行っている。

こうした研究はシステムの柔軟性という点では本研究で挙げた要件を満たさないが、行動履歴の解析手法という点では本研究で実現した解析より高度な解析を行っているものもある。これらの研究成果を参考にしながら、今後 mPATH システム上の解析部品を充実させてゆく。

7.2 ビジュアルプログラミングによる情報解析

汎用的な情報や地理情報以外の情報を対象にしたシステムにおいて、情報取得や解析手法を部品化し、ビジュアルプログラミングを実現するシステムが数多く開発されている。文献 [15] では、データフローモデルを採用したビジュアルプログラミング言語を複数紹介しながらその利点や機能、問題点を整理している。

科学技術データを対象としたデータフロー型ビジュアルプログラミング言語に、DFQL[16] がある。DFQL は、データベースに対するクエリーと、取得したデータの解析や視覚化を同一の環境で実現している。対象領域は異なるが、情報の解析だけでなく取得機能や出力機能を含めて同一のビジュアルプログラミング環境における表現を実現している点は、本研究と類似している。

商用のアプリケーションにおいても、IBM の Intelligent Miner や SAS の Enterprise Miner[17]、SPSS の Clementine[18] といったデータマイニングツールでは、データの分析結果に加えて、データの解析手法をビジュアルプログラミングの手法で設定することが出来る。本システムと同様、部品化された解析手法がアイコンとして示され、アイコン同士の流れを設定することで解析が行える。

Max/MSP[19] は、1986 年から開発されている音楽や映像のインタラクティブな編集を目的としたビジュアルプログラミング環境であり、発音アルゴリズムを自作した電子楽器の開発やアルゴリズムに基づいた自動作曲、入力した音声や MIDI イベントに対するエフェクターなど、音楽やインタラクティブメディアの製作に広く用いられている。Max/MSP はこうしたメディアに携わる人々の標準言語として利用されているほか、C 言語を利用したモジュールの製作が可能なることから広くコミュニティが形成され、ユーザによって開発されたモジュールやプログラムが公開され交換されている。

8. おわりに

本稿では、様々な行動履歴を統合的に扱い、ビジュアルプログラミングの手法で行動履歴の解析を実現する行動履歴解析システムである mPATH システムについて述べた。mPATH システムでは、行動履歴の解析手法が部品化されており、ユーザによる任意の解析部品の組み合わせで、行動履歴からさまざまな情報を読み取ることができる。mPATH システムの手法は、単一のセンサデバイスやアプリケーションを想定した従来の行動履歴の解

析手法とは異なるものであり、本システム上にさまざまな行動履歴を入力したり、さまざまな解析部品を適用することで柔軟に解析手法を構築でき、多様なアプリケーションの構築を実現する。本稿では、Javaにより実装された mPATH システムを用いて、複数のアプリケーションの構築を実現した。また、mPATH システムの性能を評価し、十分な性能を持つことを示した。

今後は、mPATH システム上で動作するのアプリケーションを充実させながら、以下に挙げる課題を中心に研究を進める予定である。

パラメータの視覚化の実現

現在、行動履歴データの入出力だけがプログラミングの対象となっているが、解析の際のさまざまなパラメータも同様に視覚化し、プログラミングを可能にする。これにより、複数の解析部品同士の連携が実現しより多様な解析が可能になる。

分散システムへの発展

行動履歴や地理情報などがネットワークに分散され保存されていたり、解析結果をネットワークを経由して他の分散アプリケーションに転送するような場合、本システム自体も個々のコンポーネントをネットワークに分散させて、分散システムとして構築したほうが柔軟なシステム構築が行えるかもしれない。今後、こうした分散システムとしての実現手法を検討する。

行動履歴の共有を目指した研究

現在、本システムでは単一ユーザの行動履歴を想定して設計されている。多人数の行動履歴を考慮した場合は、行動履歴間の比較を可能にするデータの正規化の問題や、プライバシーの問題などを考慮する必要が出てくる。今後、こうした点に留意しながら、複数人数を想定したシステムを構築について考察し、本システムに反映する。

参考文献

- [1] Weiser, M.: The Computer for the 21st century, *Scientific American*, Vol. 265, No. 3, pp. 66–75 (1991).
- [2] Abowd, G. D.: Software Engineering Issues for Ubiquitous Computing, in *proceedings of ICSE'99*, pp. 13–19 (1999).
- [3] Chen, G. and Kotz, D.: A Survey of Context-Aware Mobile Computing Research, Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College (2000).
- [4] Want, R., Hopper, A., Falcao, V. and Gibbons, J.: The Active Badge Location System, in *ACM Transactions on Information Systems, vol 10*, pp. 91–102 (1992).
- [5] Newman, W. M., Eldridge, M. A. and Lamming, M. G.: PEPYS: Generating Autobiographies by Automatic Tracking, in *Proceedings of ECSCW '91*, pp. 175–188 (1991).
- [6] Kubach, U. and Rothermel, K.: Exploiting Location Information for Infostation-Based Hoarding, in *Proceedings of the 7th ACM SIGMOBILE Annual International Conference on Mobile Computing and Networking (MobiCom 2001)*, pp. 15–27 (2001).
- [7] Abowd, G., Atkeson, C., Hong, J., Long, S., Kooper, R. and Pinkerton, M.: Cyberguide: A mobile context-aware tour guide (1997).
- [8] Wolf, J., Guensler, R. and Bachman, W.: Elimination of the travel diary: An experiment to derive trip purpose from GPS travel data, *Notes from Transportation Research Board, 80th annual meeting* (2001).
- [9] Ashbrook, D. and Starner, T.: Learning Significant Locations and Predicting User Movement with GPS, in *Sixth International Symposium on Wearable Computers (ISWC 2002)*, pp. 101–108 (2002).
- [10] Patterson, D. J., Liao, L., Fox, D. and Kautz, H.: Inferring High-Level Behavior from Low-Level Sensors, in *Proceedings of The Fifth International Conference on Ubiquitous Computing (UBICOMP2003)*, pp. 73–89 (2003).
- [11] Garmin Ltd., : Garmin eTrex Legend (2001), <http://www.garmin.com/products/etrexLegend/>.
- [12] JEIDA, : *Digital Still Camera Image File Format Standard (Exchangeable image file format for Digital Still Cameras: Exif) Version 2.1* (1998).
- [13] Environmental Systems Research Institute Inc., : *ESRI Shapefile Technical Description*, An ESRI White Paper (1998).
- [14] Environmental Systems Research Institute Inc., : ArcGIS: <http://www.esri.com/software/arcgis/index.html>.
- [15] Hils, D. D.: Visual Languages and Computing Survey: Data Flow Visual Programming Languages, *Journal of Visual Languages and Computing*, Vol. 3, No. 1, pp. 69–101 (1992).
- [16] Dogru, S., Rajan, V., Rieck, K., Slagle, J. R., Tjan, B. S. and Wang, Y.: A Graphical Data Flow Language for Retrieval, Analysis, and Visualization of a Scientific Database, *Journal of Visual Languages & Computing*, Vol. 7, No. 3, pp. 247–265 (1996).
- [17] SAS Institute Inc., : SAS Enterprise Miner: <http://www.sas.com/offices/asiapacific/japan/software/enterp.html>.
- [18] SPSS Inc., : SPSS Clementine: <http://www.spss.co.jp/product/clementine/cgp.html>.
- [19] Cycling'74, : Max/MSP: <http://www.cycling74.com/products/maxmsp.html>.