

ユビキタス情報環境における履歴を用いた機器操作支援手法

河口 信夫[†] 宮崎 俊和[†] 稲垣 康善[‡]

[†]名古屋大学大学院工学研究科 〒464-8603 名古屋市千種区不老町 1

[‡]愛知県立大学情報科学部 〒480-1198 愛知県愛知郡長久手町

E-mail: [†]kawaguti@itc.nagoya-u.ac.jp

あらまし 近年、様々な場所に情報機器が埋め込まれ、多様なサービスを実現するユビキタス情報環境が整いつつある。しかし、一般に多様な機器が利用可能になれば、必要な情報を得るための操作も増大する。ユーザに優しいユビキタス情報環境を実現するためには、可能な限りユーザの操作を最小化することが望ましい。本稿では、情報機器の利用履歴を用いて、ユーザの機器利用における操作の予測やマクロ化を行い、ユーザ支援を行う手法を提案する。本手法では、(1)個々の機器の操作が行われる度にその状況を XML データベースに保存し、(2)ユーザからの要求により、データベースから類似した状況を検索し次の操作を予測する。本手法を実際の cogma room の環境上に実現し、その有効性を確認した。

キーワード ユビキタス コンピューティング、例示プログラミング、利用履歴、XML

History based User Support for Ubiquitous Information Environment

Nobuo KAWAGUCHI[†] Toshikazu MIYAZAKI[†] and Yasuyoshi INAGAKI[‡]

[†] Faculty of Engineering, Graduate School of Nagoya University 1 Furo-Cho, Chikusa-ku, Nagoya, 464-8603 Japan

[‡] Faculty of Information Science and Technology, Aichi Prefectural University Nagakute-cho, Aichi, 480-1198 Japan

E-mail: [†]kawaguti@itc.nagoya-u.ac.jp

Abstract. Recent advancement of embedded network devices realizes ubiquitous information environment which enables person can utilize any kind of information at anytime and anyplace. However, increasing of information devices requires increasing number of user operation. To realize a smart ubiquitous information environment, it is required to minimize the number of user operation. In this paper, we propose a history based user support to minimize the number of operation. This method utilizes a XML database to record the situation and to predict the next operation. We have exemplified the method in the smart space named “cogma room”.

Keyword Ubiquitous Computing, Programming by Example, Utilization History, XML

1. はじめに

近年、様々な場所に情報機器が埋め込まれ、多様なサービスを実現するユビキタス情報環境が整いつつある。近年では、スマートルームと呼ばれる様々なセンサやコンピュータを備え付けた部屋や情報取得装置の研究[2]や、携帯電話や PHS などの基地局、GPS の位置情報等を利用したセンサネットワークの研究が盛んに行われている[3]。このような環境下で、あるユーザが様々なサービスを利用しようとする場合、利用する機器や状況によって煩雑な操作を日常的に繰り返し行わなければならないことが少なくない。例えば、バス停において次のバスの時間を調べたり、テレビの前で現在のテレビ番組表を眺めたりするといった場合、状況を考慮せずに情報を取得するためには、端末上で煩

雑なメニュー操作などが必要である。ユーザにとって快適なユビキタス情報環境を実現するためには、このような煩わしい操作をユーザに感じさせないことが求められる。

一方、特定の状況において提供されるサービスや、それに伴う操作を、一連の手順として簡便に実行できるようにシステムに組み込んでおく方法もある。しかし、多種多様な状況で求められる操作をすべてシステム上に事前に用意しておくことは不可能である。そこで、情報機器が状況に応じて次の操作を予測することや、以前の実行内容を再利用することがより有効であると考えられている[4]。

本稿では、ユビキタス情報環境がユーザの操作の履歴とその状況を学習することにより、現在の状況にお

いてユーザが行うであろう操作列を予測し、またその操作列を簡単に実行するシステムを提案する。例えば、ミーティングを行うという状況が分かっている場合、簡単な操作でプロジェクタの電源投入やスクリーンの降下、スライドの提示や資料の配布等の一連の操作を実行可能にする。

本稿の構成は以下の通りである。2節ではユビキタス情報環境におけるユーザ支援について検討する。3節では2節を踏まえユーザ支援に有効な支援手法を提案する。4節ではその手法を用いたシステムの設計を行い、5節では実装について述べる。6節で本稿をまとめ今後の課題を示す。

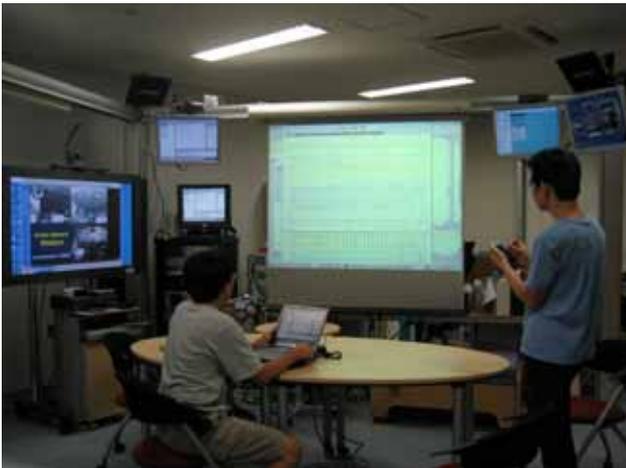


図1 スマート会議室 (cogma room)

2. ユビキタス情報環境におけるユーザ支援

ユビキタス情報環境では様々な状況で多種多様な情報機器が使用される。図1に示すような多様な機器がネットワークに接続されたスマート会議室では、プレゼンテーションを行う際には、(1)スクリーンを降下、(2)プロジェクタの電源投入、(3)ビデオ接続を切り替えといった一連の操作が必要になる。これらの操作はネットワークを通じて任意の端末から可能であるが、個々の操作を独立に行う必要がある。また個人で端末を利用する場合でも、例えばバス停や駅などで時刻表を表示したり、待ち時間に Web 上で情報収集を行ったりする。このように、「ミーティング」や、「バス停にいる」といった状況が定まれば、機器に対する操作の範囲が予測できるにもかかわらず、これまでの情報機器の操作では、すべての指示を行う必要があった。特に多数の情報機器が存在する場合や、操作メニューの選択枝が多かったり、メニュー階層が深かったりする場合に、煩雑さの度合いが高くなる。

そこで、ユーザの操作履歴と状況から意図する操作を予測したり、またユーザが例を示すことによってその例をコンピュータが覚えておき簡単な操作で再生できたりするようなシステムが望ましいと考えられる。こ

のようなシステムは予測インタフェースや例示プログラミングの技術を利用して実現できる。

予測や例示によるユーザの操作の支援を行う場合、単に予測や例示が可能であるだけでなく、ユーザがその機能を便利だと感じる必要がある。予測された操作がユーザの意図しないものであったり、例示によるプログラム作成が複雑であったりすれば予測や例示のシステムを利用する意味がなくなってしまう。そこで、予測、例示プログラミングのシステムは次の事を満たす必要がある[4]。

- a. プログラム作成が簡単である
例示プログラミングのプログラム作成において、その操作が煩雑であるとプログラム作成における手間の方が作らないときの手間に比べて大きくなってしまふ。そのためにはユーザが行った操作の履歴といった暗黙的例示からユーザが意識しなくてもプログラムを生成できる事が望ましい。
- b. プログラム実行に対するリスクが少ない
完全にシステムに実行をまかせてしまうと、ユーザが気づかぬうちに間違った実行を行ってしまうなどのリスクが生じる。このようなリスクを軽減するために、実行前の確認機能が必要である。
- c. 予測や例示の機能が邪魔にならない
予測結果が常に提示されるようなシステムでは一部のユーザや、またそれを利用しないときにそのような機能が煩わしく感じることもある。また常に予測の機能を動作させておくと予測のやり方によっては機器の処理能力の低下につながることもある。よって、予測や例示はユーザが指示したときに行われるか、行われてもユーザの邪魔にならず、また他の処理に影響を及ぼさない方法が必要である。

3. 操作履歴に基づくユーザ支援手法

本節では多数の情報機器を利用する環境で煩雑な操作を解消するためのユーザ支援手法を提案する。本手法は、分散環境における操作履歴の保存方法、操作履歴に基づく操作の予測方法、例示プログラミングを用いたマクロ作成方法の三つから構成される。

3.1. 分散環境における操作履歴の保存

ユビキタス情報環境には様々な情報機器が分散して存在する。そのため全ての情報機器の操作履歴を特定のデータベースに保存するにはネットワークを介す必要がある。ここではどのように分散した情報機器から操作履歴を收拾し、利用するかを述べる。

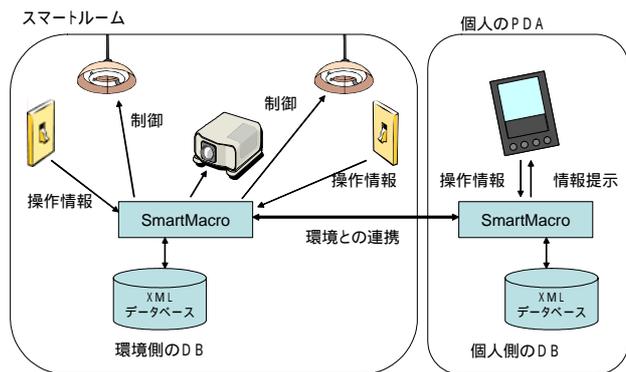


図 2 : データベースの分散配置

操作履歴の内容

一般に履歴情報は「状況」と「操作」の組で保存される。「状況」とはユーザや情報機器を取り囲む環境である。時間や場所、その他の様々な要素が考えられる。例えば「午後、会議室で」という状況には「時間」「場所」、「研究室の自分の机に座り、無線 LAN に接続して」には「場所」「ネットワーク種別」、「温度 10℃、湿度 45%、北緯 35° 43'26"、東経 137° 02'25"の位置で」では「温度」「湿度」「位置」という要素によって構成される。

これに対して、「操作」とはユーザが能動的に行った情報機器に対するふるまいである。例えば、「電灯を点ける」「プロジェクタでスライドを表示する」「OK ボタンを押す」等がある。

各情報機器の操作履歴の保存

ユビキタス情報環境で利用される情報機器は一般に、ネットワークを通じて外部から制御することが可能である。そのために情報機器は外部にコントローラを持つことが多い。このコントローラは制御情報をネットワークを介して目的の情報機器に送信するが、本手法ではその制御情報を同時にデータベースに向けて送信し、これを保存する。これによりほとんどの情報機器の操作履歴を逐次保存することができる。また本手法を実装したシステムが各コントローラを制御することで、保存されているものと同様の操作を再生することができる。

データベースの分散配置

本手法では、スマートルームや駅といった場所などの環境側に備え付けられているデータベースと、ユーザ個人の情報機器で動作しているデータベースの 2 種類を想定する。環境に備え付けられたデータベースは、環境に存在する情報機器についての操作履歴を保持している。個人の情報機器からの検索要求に対して応答し、初めてその場所に来たユーザが、履歴情報を用いて環境側のシステムを効率的に利用できる。一方、ユ

ーザのプロフィールに特化したデータを個人の情報機器のデータベースに保存することで、そのユーザのみ必要とする操作や操作を予測、再生できる(図 2)。

3.2. 操作履歴に基づいた次の操作の予測

予測は操作履歴を基に行われる。操作履歴を保存すると同時にそのときのユーザの置かれている「状況」を一緒に保存する。このとき「状況」は「場所」や「時間」といった様々な要素から構成される集合として表現される。予測はまず操作履歴を時間的にまとめた操作列に分けることから始まる。ここで操作列とはある意図をもった一連の操作を表す操作のリストである。次にその操作列から予測を指示したときの状況と同じかまたは似た状況のものを検索する。ここで同じかまたは似た状況の過去の操作履歴を得るために、現在の状況をクエリとしてデータベースを検索する。次に検索された操作列の集合の中に直前に行われた操作列を含むものを探し、あればその操作列から直前に行われた操作列の次の操作にあたる操作を取り出し、予測結果として提示する。

しかしこの方法だけではユーザの期待と異なる予測をする可能性がある。そこで答えを一つに絞らずに複数の予測結果を提示し、ユーザに決定を委ねることとする。これにより間違った予測のリスクを軽減することができる。また、提示されている操作がユーザの求めるものでは無かった場合、ユーザは他の予測を見ることができる。このとき一番もっともらしい予測から提示することが望ましい。

3.3. 例示プログラミングによるマクロ作成

本手法では、予測に加え、一連の操作列をマクロとして定義する場合についても操作支援を行う。ユーザがマクロの作成を指示すると、本手法では、指示の直前のまとまった操作を提示する。直前のまとまった操作とは、上記の予測と同じ方法で操作列として検出される。提示列の選択により、ユーザがマクロの作成を決定することにより操作列をマクロとして覚えさせることができる。本手法では、ユーザがあらかじめ例示プログラミングの開始を指定する必要がなく、操作履歴からマクロ化したい操作列を提示できるので、マクロの作成は比較的簡単である。ここで提示した操作列がユーザの意図に反する場合、操作列のまとめ方を変え再度提示しなおす機能も必要となる。

設定されたマクロを実行すると、定義された操作列が実行される。マクロはいくつも定義することができ、また複数のマクロを同じインタフェースに設定しても状況によって実行するマクロを区別することができる。

4. ユーザ支援システムの設計

本節では前節で述べた手法に基づいて、多数の情報機器を利用する環境で煩雑な操作を解消するためのユーザ支援のシステムを設計する。

履歴データ

前節で述べたように履歴データは「状況」と「操作」の組で成り立つ。そこで履歴データを l とし状況を s 、操作を a としたとき、 $l=(s,a)$ と定義する。また状況 s は様々な要素で構成され、操作 a は操作名 n と操作対象 t とその他の属性 w からなる。

ある意図をもった操作列を一つの単位として保存すると、その再現は簡単になるが、履歴情報の応用範囲を狭めることになる。履歴はできるだけプリミティブな形で保存することが望ましい。一方、ある意図をもった操作列を一つの単位として保存する事もマクロを記憶しておくという意味で必要となる。そこで、例示プログラミングによって作成されたプログラム、いわゆるマクロもデータベースに保存する。そのデータはトリガとなるボタンや状況の変化に関連付けられていて、トリガが発生したときに検索され、利用される。マクロとして保存されるデータは履歴データの「操作」の部分の集合で構成される。

4.1. 予測とマクロ作成インタフェース

操作予測や例示プログラミングの機能に必要なインタフェースとして、「予測ボタン」「ブランクボタン(マクロボタン)」「実行ボタン」の三つを定義する。

● 予測ボタン

予測ボタンを押すと次のユーザの操作を予測し、提示する。更に予測ボタンを押すと別の予測が提示される。

● ブランクボタン(マクロボタン)

ブランクボタンはマクロを定義されるボタンで、マクロの定義されたブランクボタンはマクロボタンとなる。ブランクボタンを押すとマクロ候補が提示され、更にブランクボタンを押すと別の候補が表示される。マクロボタンを押すとマクロが実行される。マクロボタンを一定時間が押しつづけてから離すとマクロが削除され、ブランクボタンになる。

● 実行ボタン

実行ボタンは予測された操作を実行する。またマクロ作成を実行する。

操作列の決定規則

操作列は、時間的に連続性のある操作を関連性があるとしてまとめたものである。間隔時間の最長時間

を定め、操作と操作の発生時刻の間隔がそれより長いかわり短いかで時間的な関連性を判断する。間隔がより短ければその操作は関連性があるとし操作列としてまとめる。

状況の比較規則

状況は状況を構成する要素の集合からなり、その要素同士の比較によりその状況が集合として等しいか、または部分集合となるかを決定できる。状況を構成する要素は要素毎に構造が異なるため、比較規則を要素ごとに決める必要がある。

例えば時間について考慮する場合、日付まで含めた同じ時刻は繰り返し現れないため、一日の中の時刻や、曜日、月の中の日付などを考慮する必要がある。また一日の中の時刻でも完全に一致することが必要な場合と、5分程度の誤差を許す場合、午前、午後といった曖昧な場合、など、多くの程度の違いが存在する。

会議などの参加者を要素として考慮する場合は、会議参加者が完全に一致する場合に加え、メンバーが足りない場合や余分な場合など、も考慮する必要がある。このように、状況は要素によって様々な比較を行う必要があり、これらを用いて状況の近接性を判断する必要がある。

5. SmartMacro システムの実装

本節では本稿で提案する手法のモバイルエージェントシステム cogma(Cooperative Gadgets for Mobile Appliances)[1]上への実装について述べる。

実装環境

ユーザの操作は主にアプリケーションの起動と終了、JavaのButtonやMenuに代表されるGUIを対象とする。cogmaはユビキタス情報環境における情報機器の連携アプリケーションを容易に実現する目的で作成されたミドルウェアである。cogmaを利用することにより、cogma roomに設置されたプロジェクタ、照明、スクリーンなどをネットワークを介して容易に操作できる。cogma上のアプリケーションをcodgetと呼ぶ。

履歴データベース

履歴情報を保存するXMLデータベースとしてXindiceを用いる。今回のような場合、状況を構成する要素によってその情報の構造が異なっているため一定の型を持つデータ構造では扱いきれない。また将来的に全く新しいタイプの情報を、状況の構成要素として後から付加することを考えると、拡張性の高いデータ構造が望ましい。XMLデータベースは異なる文書構造をデータベースに保管し、それらをまたいで照会することができるため、その点で適切である。

図 3 に今回使用する履歴データの構造を示す。

```
<data date="20xx/xx/xx xx:xx:xx"
  milidate="ミリ秒">
  <action name="操作名"
    target="アプリケーション 1 の名前:ID"
    属性名 1="属性値 1" 属性名 2="属性値 2">
  </action>
  <status>
  <member num="2">
    <mynode name="自端末の名前" />
    <node name="他端末 1 の名前" />
  </member>
  <registered num="2">
  <codget
    name="アプリケーション 1 の名前と場所">
    アプリケーション 1 の名前:ID
  </codget>
  <codget
    name="アプリケーション 2 の名前と場所">
    アプリケーション 2 の名前:ID
  </codget>
  </registered>
  </status>
</data>
```

図 3：履歴情報を表す XML データ構造

一つの履歴情報 data は action タグと status タグからなる。また属性として、時刻の情報をもつ。

action タグは操作の情報のタグである。name という属性は操作名を表し、target という属性は操作の対称となるアプリケーションの名前と ID を表す。また他の属性はアプリケーションの作成者が自由に設定できる。ユーザの能動的な操作でなく、状況の変化に伴うイベントは同じようにして action タグの代わりに change タグというタグで書き込まれる。status タグは状況を構成する要素のタグを持つ。上の例ではネットワークのメンバをあらわす member タグと現在起動されているアプリケーションを表す registered タグがある。基本的に各状況の構成要素のタグはそれぞれ異なる構造を持っている。member タグではネットワークのメンバの数を属性として持ち、mynode タグと node タグを持つ。この二つのタグは端末名を表す属性を持っている。registered タグもほぼ同様である。

モジュールの構成と実装

実装は「各機器の操作や状況の変化の記録を行う」モジュールと「状況を監視し、予測やマクロの作成、再生を行う」モジュールが中心となる。前者は履歴保存モジュール、後者は予測/例示モジュールと定義する。

これらの二つのモジュールはネットワークのメンバの追加や離脱といった状況の変化を感知しそれに対応する必要がある。cogma ではその機能を LinkMonitor と呼ばれるインタフェースで提供している。

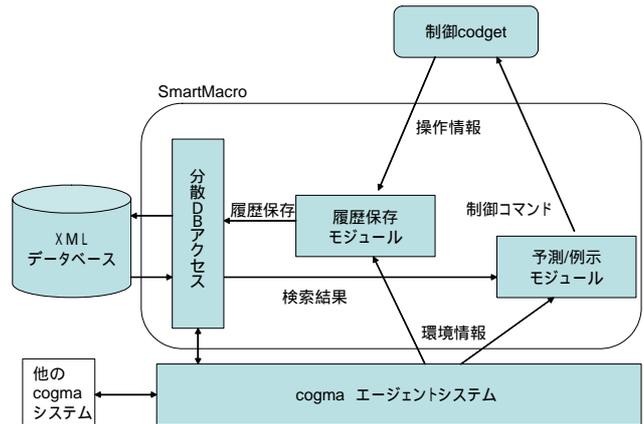


図 4：SmartMacro システムの構成

本システムの構成を図 4 に示す。操作対象となる機器には、機器を制御するための制御 codget が存在する。この codget はユーザから各機器への指示に応じて、履歴保存モジュールへ、操作情報を送る。これに対し、予測/例示モジュールはデータベースを用いて、類似の状況からの予測や、マクロの構築を行い、制御 codget へ制御コマンドを送付する。

以下に各モジュールの機能を示す。

履歴保存モジュール

cogma エージェントシステムからの情報や制御 codget からのイベントがある毎に、XML データベースへ状況を保存するモジュール。この場合イベントとは、オンラインメンバの追加と離脱、システムに対する codget の登録と解除、システムに登録されている制御 codget からの操作情報である。本システムを実装した codget はデータベースに登録したい操作の詳細を記述しておかなければならない。操作は action と呼ばれ、target と他の属性を持つ。target は操作の対称となる codget の名前と ID である。属性は複数もつことが許されており、属性名と属性値の組で構成され、codget 作成者が自由に決定できる。またそれと同時に状況を構成する要素としてネットワークのメンバや現在登録されている codget の情報も保存する。

予測/例示モジュール

ユーザが予測やマクロの作成を指示したときに、データベースを検索し、予測やマクロの作成を行う。

作成されたマクロを記憶し、そのトリガが引かれたとき対象となっている制御 codget に対し制御コマンドを送付する。操作は対象となっている制御 codget の操作再生を行うメソッドを呼び出すことによって行われる。本システムを実装した制御 codget にはそのメソッドを実装し、呼び出されたときの処理を書かなければ

ならない．また予測された操作の実行も同じように行われる．

分散データベースアクセス

環境に属しているデータベースや個人に属しているデータベースを透過的に利用するために、分散データベースアクセスを行うモジュールが必要である．本モジュールはノードの周辺に存在する SmartMacro システムを検知し、利用するデータベースを自動的に把握する．また、履歴情報の保存や検索要求を各データベースに送ると同時に、結果を統合して予測/例示モジュールに返す．

また XML データベースの検索には XPath を用いる．オンラインメンバが TSUKSI (自端末) と Kblue の状況となっている履歴情報を検索するときは

```
/data[status[member[@num="2"]  
[mynode="TSUKUSI"]][node="Kblue"]]]
```

となる．このようなクエリを用い、履歴から同じ状況を探す．

図 5 に SmartMacro システムが動作している動作画面を示す．この画面上には仮想的な電灯を制御するライトコントローラと、仮想的なライトエミュレータが二つ、またプロジェクタとマトリクススイッチャの操作パネルが表示されている．これらの制御 codget を用いることにより、任意の操作を記録・再生することができる．また左上にあるのが SmartMacro の構築用インタフェースであり、exec ボタンや guess ボタンで各操作を予測、再生することができる．特にボタンの横にはマクロを定義するためのリストが表示されており、マクロ構築ボタンを利用することによって、システムが推測した操作列が表示され、マクロ定義を容易に行うことができる．

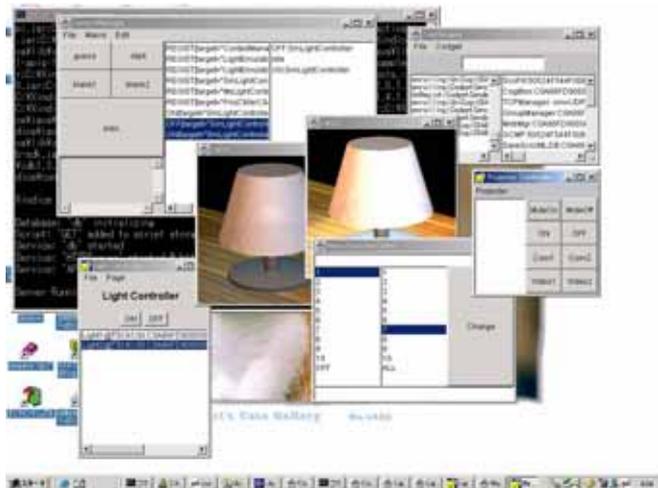


図 5 : SmartMacro システムの実行画面

6. まとめ

本稿ではユビキタス情報環境において、煩雑な操作を解消し、より快適な環境を実現するために、機器の操作履歴に基づく予測インタフェースや例示プログラミングを用いるユーザ支援手法を提案し、モバイルエージェントシステム cogma 上に実装した．

特に本システムでは、状況を記録するデータベースを分散的に配置可能とすることにより、ユーザがはじめて利用する場所でも、過去のユーザの利用履歴を用いることによって、快適な利用環境を実現することができる．

現在の実装では、状況間の比較や予測のアルゴリズムについては十分な考慮がなされておらず、任意の状況において求める予測を得ることはできない．今後、各状況における比較の手法や、予測アルゴリズムの高精度化が求められる．

また提案した手法により、ユーザ端末から複数の情報機器を利用する一連の操作を簡単な操作でマクロ化し、実行すること可能となった．今後は被験者による評価実験などにより本手法の有効性を評価する必要がある．

文 献

- [1] 河口信夫, 稲垣康善: "cogma:動的ネットワーク環境における組み込み機器間の連携用ミドルウェア", 情報処理学会コンピュータシステム・シンポジウム, pp.1-8(Nov.2001)
- [2] 中村嘉志, 伊藤日出男, 西村拓一, 山本吉伸, 中島秀之: "無電源小型通信端末 CoBIT による近距離情報支援の実現", 情報処理学会知的都市基盤研究グループ研究報告, 2002-ICII-3, pp.1-7 (2002)
- [3] 岩谷晶子, 西尾信彦, 村瀬正名, 徳田英幸, "ごましお: アドホックセンサネットワークにおけるノード位置決定方式", 情報処理学会モバイルコンピューティングとワイヤレス通信研究会 Vol.2001(108), pp.23-30, (Nov. 2001)
- [4] 増井俊之: "予測/例示インタフェースシステムの研究", 学位論文, 東京大学, (Nov.1997)
- [5] Henry Lieberman: "Your Wish Is My Command Programming by Example", Morgan Kaufmann Publishers (2001)
- [6] 古市悠, 志和木愛子, 岩井将之, 徳田英幸: "Sticky Editor System:基盤型実世界指向アプリケーション", マルチメディア,分散,強調とモバイル(DICOMO2003)シンポジウム論文集, pp.649-652, (Jun.2003)
- [7] L.Barkhuus and A.Dey: "Is Context-Aware Computing Taking Control away User? Three Levels of Interactivity Examined", Ubicomp2003, pp.149-156, (Oct.2003)