

Spot & Snap : DIY Smart Object Serviceを実現する センサノードと日用品の関連付けインタラクション

米澤 拓郎[†] 榊原 寛[†] 中澤 仁[†]
高汐 一紀^{†,††} 徳田 英幸^{†,††}

本論文は、ユーザ自身が日用品にセンサノードを取り付け利用する DIY (do-it-yourself) Smart Object Services と、それを実現するための、センサノードとモノの情報を関連付けインタラクション Spot & Snap を提案する。センサノードを取り付けた日用品 Smart Object を利用したサービスでは、センサノードと日用品の意味的な関連付けが必要である。しかし関連付け作業は複数のステップからなり、一般家庭のユーザが行うには困難で煩わしい作業である。本論文ではこの問題を解決するため、スポットライトが取り付けられたカメラを利用した関連付け手法 Spot & Snap を考案・実装し、Spot & Snap を利用したアプリケーションを構築した。また実装システムの性能とユーザビリティについて実験を行って評価し、有用性を示した。Spot & Snap は、センサノードが取り付けられたモノにスポットライトを照射するという 1 つのインタラクションで容易に関連付けを行える。また同時に、アプリケーション開発者にとって Smart Object Service を実現するアプリケーションフレームワークを提供する。

Spot & Snap: An Interaction for Associating Sensor Nodes and Everyday Objects to Realize DIY Smart Object Services

TAKURO YONEZAWA,[†] HIROSHI SAKAKIBARA,[†] JIN NAKAZAWA,[†]
KAZUNORI TAKASHIO^{†,††} and HIDEYUKI TOKUDA^{†,††}

This paper introduces the concept of DIY (do-it-yourself) Smart Objects Services that enables non-expert users to apply smart object services for everyday life. When using smart object services, a semantic connection between sensor node and domestic objects must be made before services function properly. At home, however, professional assistance with such installation may be either unavailable or too costly. Spot & Snap solves this problem by easing of such association with use of a USB camera and an LED spotlight. With Spot & Snap, nonexpert users can register their belongings to preferred services without experts by taking their pictures. It also provides application framework to create various smart object services. This paper describes the prototype implementation of Spot & Snap and a range of its applications. In addition, it evaluates Spot & Snap through capability and usability experiments.

1. はじめに

ユビキタスコンピューティング環境を実現するための要素技術として、超小型無線センサノード技術が急速に発展している。本論文ではセンサノードを、照度センサ、温度センサ、加速度センサ等の複数のセンサを搭載し、取得したセンサ値を無線ネットワークに送信する小型計算ノードと定義する。センサノードから

得られる照度、温度、加速度等のデータを処理し、人や環境の状態を抽出することにより、異常検知や機器の自動制御といった実世界の状況に即したサービスを構築できる。現在、センサノードを日用品（以下、モノ）に取り付け、モノの状態を扱うアプリケーションの研究が活発である。たとえばセンサノードをコップに取り付け、飲み物の温度を感じし冷める前にユーザに通知する MediaCup¹⁾ や、身の回りのモノに取り付けたセンサノードを利用し、忘れ物をユーザに通知する SPECS²⁾ 等があげられる。我々の生活はモノとのインタラクションに溢れており、生活を支援する目的を持つユビキタスコンピューティングにおいて、モノの状態を利用したサービスの必要性は今後よりいっそう高

[†] 慶應義塾大学大学院政策・メディア研究科
Graduate School of Media and Governance, Keio University

^{††} 慶應義塾大学環境情報学部
Faculty of Environmental Information, Keio University

まっていくと予想される．ここでは，センサノードを取り付けたモノを Smart Object と呼び，Smart Object を利用して実現されるサービスを Smart Object Service と呼ぶ．

Smart Object Service を実現するアプリケーションは，センサ情報が示すモノを判別するために，どのセンサノードがどのモノに取り付けられているのか，という意味的な関連付け情報を知っておく必要がある．各センサノードには識別子としてセンサノード ID が割り当てられており，アプリケーションはセンサノード ID とモノの情報の関係性を保持することになる．前述した MediaCup や SPECs 等の研究例では，この関連付けはコンピュータの知識を持つ専門家によって行われることが想定されている．現在，この作業は (1) センサノード ID の特定，(2) 特定した ID とモノの情報の関係付け，という過程を経て行われる．新たにモノをアプリケーションの対象にする際や，センサノードを取り替える際にもこの作業を行わなければならない．よって家庭のユーザがこれらのサービスを利用するには，導入時，メンテナンス時において煩わしい作業を必要とし，時間的負担へとつながる．

我々は，サービスの対象をユーザ自身が登録し，利用できるサービス DIY (do-it-yourself) Smart Object Service を提案する．本論文では，DIY Smart Object Service を実現するために，センサノードとモノの容易な関連付けインタラクション Spot & Snap を考案した．Spot & Snap では，ユーザは関連付けの対象物 (センサノードを取り付けたモノ) をスポットライトで照射するだけでよい．照射と同時に撮影を行い，システムが自動取得したセンサノード ID と画像を関連付ける．「スポットライトで照射されたモノとセンサノードが結び付く」ということを意識する以外，ユーザにセンサノード ID 等の専門知識はいっさい要求しない．Spot & Snap を利用することで，モノの画像を利用した視覚的なサービスの構築が可能である．

本論文では，2 章で DIY Smart Object Service の概念について説明し，モノとセンサノードの容易な関連付け手法の必要性を述べる．そして，本論文で提案する Spot & Snap の手法とその応用の可能性を 3 章で述べる．また Spot & Snap を実現するための実装システムを 4 章で述べ，その応用アプリケーション例を 5 章で紹介する．そして，6 章で Spot & Snap の評価を行い，改良点を明らかにする．7 章で関連研究を述べ，8 章で今後の課題を整理し，9 章で結論を述べる．

2. DIY Smart Object Service

本章では DIY Smart Object Service の概念と要件，実現にあたっての問題点と解決のためのアプローチを述べる．

2.1 機能要件

近年，我々のライフスタイルは多様化し，人によって扱う日用品や，必要とするサービスは千差万別である．本論文で提案する DIY Smart Object Service は，家庭やオフィス環境を対象として，1 つのプラットフォーム上で多数のサービスを実現し，そのサービスの対象としてユーザ自身の持ち物が利用できることを目標とする．MediaCup¹⁾ や DigiClip⁹⁾ 等の Smart Object Service では，あらかじめセンサノードが備え付けられたモノを利用することが想定されており，ユーザの身の回りにすでに存在し日常的に扱われるモノを対象としていない．これに対し DIY Smart Object Service は，ユーザ自身がサービスやその対象となるモノを選べるため，ユーザのニーズに柔軟に対応できることを目標とする．DIY Smart Object Service を実現するために，以下の 2 点を満たす環境の構築が必要である．

● Service Selection の実現

ユーザはインターネットを通じ，必要とするサービスを実現するアプリケーションを手に入れることができる．本論文では，家庭に存在する PC と，ユーザがホームセンタ等であらかじめ購入する汎用的なセンサノードを，サービスを実現するハードウェアプラットフォームと想定する．このプラットフォームで動作するアプリケーションを，開発者が容易に作成できるためのフレームワークが必要である．

● Universal Configuration の実現

ユーザは自身の持ち物にセンサノードを取り付け，容易にサービスに登録することができる．Smart Object を扱うサービスでは，どのセンサノードがどのモノに取り付けられたかという情報が重要となる．つまり，センサノードを区別するセンサノード ID と，モノを示す情報 (モノの属性情報や状態情報等) との関連付けが必要である．ユーザは，センサノードをモノに取り付けると同時に，サービスに対しその関係性を容易に登録できる環境が必要である．

2.2 問題点

センサノードとモノの関連付けは，「センサノード ID の特定」と「センサノード ID とモノの関連性入力」の 2 つの作業からなる．この作業は複雑さをともなうため，従来は PC の知識を持つプログラマが行ってきた．ユーザが関連付け作業を行う際の問題点は，

大きく以下の2つである。

- センサノード ID の特定が困難

センサノードにはディスプレイ等の表示インタフェースがないため、センサノード ID を直接特定することができない。そのため専門のツールを使うか、センサノードがネットワークに送信するセンサデータのバケットダンプをもとに ID を推測しなければならない。環境内に複数個のセンサノードが同時に存在した場合には、1 つを特定するのはさらに困難となる。

- 手作業で情報を登録する作業負荷

センサノード ID の特定後、モノの情報とセンサノード ID の関連性をアプリケーションに入力する必要がある。多数のモノを扱うアプリケーションでは、モノの数に比例して作業量が増えてしまう。

上記の作業は、PC の扱いに不慣れなユーザが行うことは困難である。大量の ID 特定・入力作業を必要とするため、たとえ行えたとしても時間的拘束は避けられず、煩わしさは免れない。

2.3 アプローチ

我々は、問題点で述べた関連付け作業を容易にし、DIY Smart Object Service を実現するために、Spot & Snap を提案する。Spot & Snap は、Universal Configuration を実現するインタラクションであると同時に、Service Selection を実現するためのアプリケーションフレームワークを提供する。いい換えれば、Spot & Snap は、センサネットワーク層とアプリケーション層の間に、どのセンサノードがどのモノに取り付けられているかという実世界情報をバーチャル空間に対応させるためのアソシエーション層と、その情報をアプリケーションが扱うための API を提供することとなる。Spot & Snap による Smart Object Service の利用モデルを図 1 に示す。ユーザは、(A) 必要とするアプリケーションをインターネットからダウンロードし、起動する。そして (B) 自分の持ち物にセンサノードを取り付け、(C) Spot & Snap を利用してセンサノードとモノを関連付け、アプリケーションに登録する。以上で、(D) 自分の持ち物を利用した Smart Object Service を利用できる。次章で、Spot & Snap につい

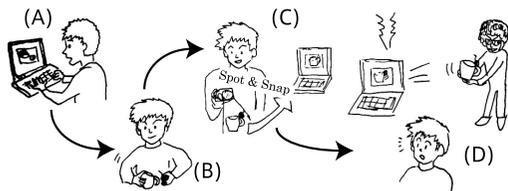


図 1 Spot & Snap に基づくサービスモデル
Fig. 1 Spot & Snap based service model.

ての詳細を述べる。

3. Spot & Snap インタラクション

本章では、Spot & Snap インタラクションの概念と特徴について述べる。

3.1 Spot & Snap の動作

Spot & Snap インタラクションは、センサノード ID の特定、およびセンサノード ID とモノの関連性入力を、スポットライトが取り付けられたカメラによって実現する。その際に必要な動作を図 2 に示す。

ユーザはまず、センサノードを任意のモノに取り付ける。そして、スポットライトを ON にし、センサノードとモノに光を 1-2 秒間照射する。スポットライトを OFF にすると、カメラはモノの撮影を行う。Spot & Snap システムは、Spot の動作により環境に存在するセンサノードのうち、スポットライトによる照射を行ったのと同時刻に照度センサが最高値を示したセンサノードを特定し、その ID を取得する。そして Snap の動作により取得したセンサノード ID と撮影したモノの画像を関連付け、Smart Object Service を実現するアプリケーションに登録する。Spot & Snap は、スポットライトを照射するという単純な動作で、ユーザがセンサノード ID とモノの画像を関連付けることを可能にする。

3.2 センサノード ID 特定アルゴリズム

多数のセンサノードが存在する環境下で、Spot & Snap は 1 つのセンサノードの照度を一定時間だけ最高値にすることで他センサノードと区別し、センサノード ID を特定する。なお、前提としてセンサノードはある一定の間隔でシンクノードに自身の ID とセンサ値を含んだバケットを無線を用いて送信し、各コンピュータはそのデータをシンクノードがネットワーク内にブロードキャストすることで受け取ることができる。現在市販されるセンサノードの多くがこの機能を有して動作しているため、前提としては適切と考えられる。

スポットライトのスイッチを ON にした時間を T_s としたとき、 $T_s < t_s < T_s + N$ を満たす時間 t_s に照度値が最高値を示し始めたセンサノードの集合を S_1



図 2 Spot & Snap における関連付け動作
Fig. 2 Mapping interaction of Spot & Snap.

とする。また、スポットライトのスイッチを OFF にした時間を T_e としたとき、 $T_e < t_e < T_e + N$ を満たす時間 t_e に照度値が最高値を示し終えたセンサノードの集合を S_2 とする。このとき、積集合 $S_1 \cap S_2$ の要素数が 1 の場合、その要素をスポットライトによって照射されたセンサノードの ID と判断する。もし要素数が 2 以上であれば、センサノード ID の特定が失敗したと判断する。スポットライトを照射してからそのセンサノードのセンサデータを PC が受け取るまでには、センサノードのデータ送信間隔とネットワークの遅延を考慮する必要がある。そのため、時間 T_s および T_e から許容時間 N を設けている。この N は、それぞれのセンサノードのデータ送信間隔によって異なる。センサデータの packets 落ちも考慮し、システムの冗長性を高めるため、過去 5 回のセンサデータの平均受信間隔を i 秒としたとき $N = 2i$ とする。

3.3 主な特徴

Spot & Snap インタラクションの主な特徴と応用の可能性を以下に記述する。

- 照度センサ値を利用したセンサノード ID の特定
Spot & Snap では、スポットライトを照射してセンサノードに搭載された照度センサの値を特徴付け、そのセンサノード ID を特定する。現在開発されている多くのセンサノードにあらかじめ照度センサが搭載されているため、Spot & Snap は多様なセンサノードを対象とできる。またセンサノードに対し、特別な実装はいっさい必要ないため、既存システムとの融和性が高い。照度センサが備わったセンサノードはすべて Spot & Snap の対象とできる。

一方、照度の代わりに加速度や温度センサの値を特徴付ける方法も考えられるが、以下の点で現実的でない判断した。加速度を特徴付けるためにはセンサノードを振ったりついたりするアクションが必要となる。しかしこれではモノの情報を取得するための撮影と一体化できず、インタラクション数が増えてしまう結果となる。また温度センサの値を特徴付ける方法では、ドライヤ等でセンサの温度を上げる方法が考えられるが、温度の上昇に時間がかかったり、ドライヤの小型化が難しく、カメラと一体化した小型装置にはなりえない。センサノード ID が記載された二次元バーコードを、工場出荷時にセンサノードに取り付けカメラで読み取る方法も考えられる。しかしセンサノードを開発しているすべてのベンダに二次元バーコードの作成と取り付けを行ってもらうことは現実的ではない。さらに、モノに取り付けられる小型センサノードには、二次元バーコードを取り付けるための物理的スペース

が存在しない。

一方、照度センサは環境光の影響を受けやすく、そのロバスト性について検証する必要がある。この評価は 6 章に示す。

- スポットライトによる直感的インタフェース

1 点を明るく照らすスポットライトは、元々舞台において観客の注目を集めるための装置である。Spot & Snap においてスポットライトは、その指向性により対象物のみを正確に関連付ける補助、そしてユーザに対する視覚的フィードバックの役割を持つ。すなわち、視覚的に分かりやすい直感的操作を可能にする。

- 特別なインフラが必要ない

Spot & Snap は、特別なサーバや高価な機器を必要とせず、家庭にあるコンピュータに Spot & Snap で用いるデバイスをつなげるだけで実現できる。このデバイスは、WEB カメラにスポットライトを取り付けたものを想定しており、ユーザに金銭的負担を与えない。よって一般家庭での利用が可能と考えられる。またスポットライトは、デジタルカメラで撮影する際に焦点を合わせたり暗い場所を明るくするための補助光として使用されており、カメラに標準で装着されていることが多い。よって将来的には携帯電話やデジカメを利用して実現できる。

- 画像を利用した関連付け

Spot & Snap では、センサノード ID を関連付ける対象としてモノの画像を利用している。モノの画像によりアプリケーションは、視覚的なサービスを提供できる。一般的に人は視覚器官を介してモノを知覚し、ある印象を脳で形成する。そしてその印象を過去の記憶と結び付け、モノの名前を認識し、性質を理解する¹⁰⁾。「コップ」という名前だけでは、どのコップを示しているのか分からないように、名前や属性情報だけでモノを判別するのは難しい。このため、ユーザにとって、サービスに登録されているモノの表現手法として画像を利用することは有効であると考えられる。また、本論文においては対象としないが、オブジェクト認識の様々な技術を適用することで、撮影した画像からモノの名前等のメタ情報を抽出し⁵⁾、より多様なサービスが構築できることが考えられる。本論文では、画像のみで実現できるサービスについて検討し、その例を 5 章で述べる。

4. 実装システム：uAssociator

uAssociator は、Spot & Snap インタラクションを実現するために実装されたプロトタイプシステムである。本章では、uAssociator のハードウェアとソ



図 3 uAssociator Camera
Fig. 3 uAssociator Camera.



図 4 uAssociator Camera を用いた関連付けの様子
Fig. 4 Mapping interaction with uAssociator Camera.

ソフトウェアについて説明する。なお、ソフトウェアは JAVA で実装されており、画像処理には JAVA Media Framework (JMF) を利用している。

4.1 ハードウェア

uAssociator で利用するデバイスとして、スポットライトを WEB カメラに取り付けた uAssociator Camera を試作した(図 3 参照)。スポットライトにはオプトエレクトロニクス社の小型 LED Spotlight を、WEB カメラには ELECOM 社の 30 万画素 WEB カメラをそれぞれ用いた。またスポットライトの ON/OFF を検知するために、小型の無線マウスをスポットライトの下に取り付け、認識できるようにした。uAssociator Camera を利用した関連付けの様子を図 4 に示す。uAssociator Camera は小型で、片手で操作できる。また、センサノードとして Karlsruhe 大学 TecO 研究所で開発された μ Part⁴⁾ と、CrossBow 社が販売している MOTE⁶⁾ を用いた。それぞれ照度・加速度・温度センサが搭載されており、現在入手可能なセンサノードとしては小型である。実装で使用したセンサノードは、センサデータ送信間隔を 0.5 秒間隔とした。

4.2 ソフトウェア

uAssociator のソフトウェア構成図を図 5 に示す。

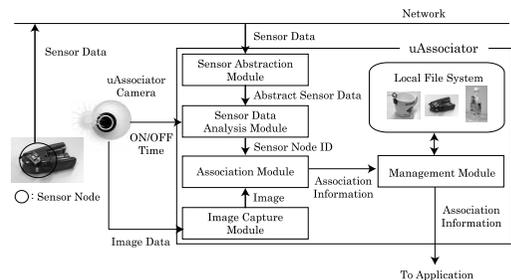


図 5 uAssociator のソフトウェア構成図
Fig. 5 Software architecture of uAssociator.

uAssociator では、多様なセンサノードを用いた関連付けをサポートしている。Sensor Abstraction Module では、環境内に存在するセンサノードを動的に発見し、統一的なインタフェースにより抽象化する¹¹⁾。これにより、システム内の他の部分は、特定のセンサプラットフォームに依存せずに動作する。Sensor Data Analysis Module は、センサ抽象化モジュールが抽象化した照度センサデータを解析し、3.2 節で述べたアルゴリズムによってスポットライトが照射されたと判断したセンサノード ID を Association Module に渡す。Image Capture Module はスポットライトが点灯したことを検知し、モノの画像を撮影する。取得された画像は、Association Module に渡される。Association Module は受け取ったセンサノード ID と画像を関連付ける。関連付けられた情報は Association Information として Management Module に渡される。Management Module は Association Information をアプリケーションに渡す。

また、将来ユーザが一度関連付けたモノを他のアプリケーションに登録できるように、Management Module は Association Information を uAssociator Image File として Local File System 内に保存する。uAssociator Image File は JPEG 形式の画像ファイルとして保存され、センサノード ID 等の関連付けられたメタデータは、JPEG のユーザコメント領域に保存される。メタデータの一例を図 6 に示す。データのフォーマットについては、XML を適用した。将来の拡張のため、モノの名前を表すフィールド、アプリケーションが利用できるコメントフィールドが用意されている。ユーザは uAssociator Image File をアプリケーションに Drag & Drop することで、Association Information としてモノを登録することができる(図 7)。

アプリケーション開発者は、Smart Object のセンサデータの処理方法とその処理に対応したアクチュエー

```

<?xml version="1.0" encoding="shift_jis" ?>
<association_info>
  <node_type> uPart </node_type>
  <node_id> 1.2.3.4.0.1.0.102 </node_id>
  <name> Unnamed Object </name>
  <created_time>
    2006/05/16 21:43:54
  </created_time>
  <comment> </comment>
</association_info>

```

図 6 uAssociator Image File に保存されるメタデータの記述例
Fig. 6 Sample meta data stored in uAssociator Image File.



図 7 Drag & Drop によるモノの登録
Fig. 7 Register of object by Drag & Drop.

```

public class ExApp implements AssociationListener{

  public boolean processing(){
    getData(sObject.getType(), sObject.getId());
    if(EVENTOCCURS) return true;
    return false;
  }

  public void actuating(){ Light.on(); }

  public void spotAndSnap(SmartObjectEvent e){
    sObjectList.addElement(e.getSmartObject());
  }
}

```

図 8 アプリケーションのプログラム例
Fig. 8 Sample program of application.

ションを記述することで Smart Object Service を提供するアプリケーションを構築できる。アプリケーションのプログラム例を図 8 に示す。AssociationListener クラスは、イベントリスナとして Spot & Snap の動作を監視し、関連付けが行われた場合にそのセンサノード ID とモノの画像、センサノードの種類等の情報を内部オブジェクトとして保持する SmartObject クラスを取得する。

5. アプリケーション

Spot & Snap の有効性を確認するため、いくつか

表 1 uAssociator および各アプリケーションのプログラム行数
Table 1 Amount of code lines of uAssociator and applications.

uAssociator	uViewer	uAlert	uCare	uServiceMaker
1,321	1,684	1,342	2,176	3,071

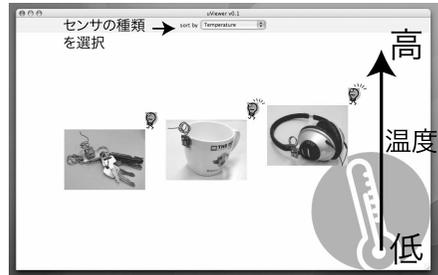


図 9 uViewer のプロトタイプ
Fig. 9 Prototype of uViewer.

の応用アプリケーション例を試作した。uAssociator および各アプリケーションのプログラム行数を表 1 に示す。以下に、それぞれのアプリケーションについて説明する。

5.1 uViewer

uViewer は、モノの状態を記録し、その履歴を表示することによって、ユーザの日用品管理を補助するアプリケーションである(図 9)。たとえば、モノの環境情報の変化を履歴として可視化することで、そのモノの使用頻度を知ることができる。またリアルタイムにモノの状態を可視化するので(たとえば動いているモノの画像が揺れる等)、目の届かないモノでもその状態を把握できる。

5.2 uAlert

uAlert は、引き出しやタンスの中に入った預金通帳や日記等、大切なモノが取り出される可能性が生じると、即座にそのことを通知するアプリケーションである。登録されたモノが入った引き出しが開けられると、その振動と外界の光を検知しアラームを鳴らす。同時に、モノの画像をユーザの携帯端末に「取り出される可能性がある」と通知する。ユーザは自分の持ち物を画像として判断できるため、どのモノが取り出されようとしているのか、視覚的に判断できる。

5.3 uCare

uCare は、1 人暮らしの高齢者を見守るアプリケーションである。高齢者が普段利用する日用品や家具を uCare に登録すると、uCare はそれらの使用状況を家族や介護者に伝える。家族は高齢者がどのモノを使っているか、という情報をモノの画像を通じて把握することができる(図 10)。家族は、高齢者が普段

使っているモノがその日は使われなかったといった状況を uCare を通じて認識し、高齢者を気遣って電話をかけるといったことが想定でき、コミュニケーションの活性化も想定できる。類似したサービスに象印社の iPot²⁰⁾ があげられる。iPot はポットの使用状況を家族にメールで通知するサービスだが、uCare は Spot & Snap を利用することでポット以外の様々な日用品もサービスの対象とできる。そのため、個人の生活スタイルに合わせたサービスが提供できると考えられる。

5.4 uServiceMaker

uServiceMaker は、ユーザが簡単にモノの状態を利用したサービスを構築できるアプリケーションである(図 11)。ユーザは、モノがある状態をイベントとして定義し、定義したイベントを様々なアクチュエータ(アラームを鳴らす、メールを送信する、情報家電を制御する等)のトリガとして利用することができる。この設定は、uServiceMaker の GUI 上に画像として表示される Smart Object に対してイベントを設定し、任意のアクチュエーションを示すアイコンと線をつなぐことで実現できる。たとえばユーザが椅子に座ると、椅子に取り付けられた圧力センサがユーザの体重を検知し、机のライトを点灯させる等のサービスをユーザ自身が構築できる。

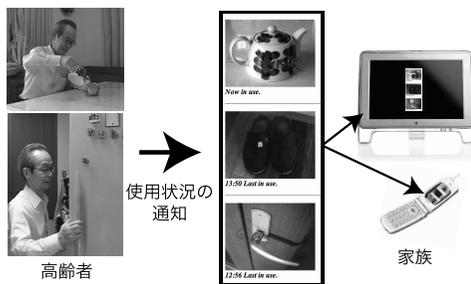


図 10 uCare の動作概要
Fig. 10 Overview of uCare.



図 11 uServiceMaker のプロトタイプ
Fig. 11 Prototype of uServiceMaker.

6. 評価

本章では、本論文で提案する Spot & Snap および実装システム uAssociator の性能とユーザビリティを評価する。性能評価として、複数の環境下で uAssociator の使用実験を行い、結果を考察する。ユーザビリティの評価として、複数の被験者に uAssociator を使った実験を行い、結果を考察する。

6.1 性能評価

uAssociator の性能を評価するため、様々な環境でどの程度センサノード ID が特定可能か、実験を行った。実験には μ Part⁴⁾ と DOT⁶⁾ の異なる 2 種類のセンサノードを利用した。晴天の昼間に部屋の中心にある机の上(部屋のすべての蛍光灯をつけた状態)、TV の前、卓上ライトの下、窓際のそれぞれの場所で関連付け作業を 50 回ずつ行い、センサノード ID の特定が成功したかどうか判定する。またそれぞれの場所には関連付けで用いるセンサノードとは別のセンサノードを計 20 個設置しておき、複数のセンサノードの存在下で実験を行った。センサノード ID 特定の成功率を表 2 に示す。

机の上、TV の前では μ Part、DOT いずれも高い確率で特定が成功した。失敗したのは、主にセンサデータの急激なバケット落ちが原因であった。他センサノードの存在の影響による ID 特定エラーは発生しなかった。一方、卓上ライトの下では μ Part の ID の特定ができなかった。これは μ Part に搭載された照度センサの方が DOT の照度センサよりも光に敏感に反応するセンサで、卓上ライトの下では μ Part の照度センサがつねに最高値を示したためである。一方、窓際では μ Part、DOT いずれも照度が最高値となっており、成功しなかった。

これらの結果から、uAssociator は本研究の対象環境であるリビングルームや寝室等、屋内の大部分では利用できることが分かった。一方、窓際で利用するにはカーテンを閉める等の対応が必要である。また、 μ Part と DOT の比較より、様々な場所で uAssociator を利用するためにはできるだけ高い照度レベルの差異を検知できる照度センサが求められることが分かった。

表 2 場所ごとのセンサノード ID 特定の成功率

Table 2 Success rate of identifying sensor node ID according to each place.

	机の上	TV の前	卓上ライトの下	窓際
μ Part	88%	90%	0%	0%
DOT	96%	94%	92%	0%

6.2 ユーザビリティ評価

ユーザビリティの指標¹³⁾を表す学習しやすさ (Learnability), 効率性 (Efficiency), 記憶しやすさ (Memorability), エラーがあるかまたは起こした際の回復 (Error Handling), ユーザの主観的満足度 (User Satisfaction) の5つに関して, 実験を行い評価した.

6.2.1 実験

Spot & Snap により直感的な関連付けが行えることに対する評価を行うため, 実験を行う.

本実験の被験者は, uAssociator の利用経験がない10代から20代の学生13名(被験者A-Mとする)である. そのうちコンピュータをほとんど使わないコンピュータ初級者が3名(被験者A-C), コンピュータを日常的に使っているコンピュータ中級者が6名(被験者D-I), OSやネットワークの知識, プログラミングスキルのあるコンピュータ上級者が4名(被験者J-M)である. 実験環境となる机の上にはuAssociator CameraとuAssociatorシステムが動作するPC, センサノードとセンサノードを取り付けるためのコップが置かれている. 実験は, 部屋のすべての蛍光灯がついた状態の研究室で, 昼間から夕方にかけて行われた. 実験の手順は, uAssociatorに関する簡単なオリエンテーション(uAssociatorの目的, 使い方, 実験の諸注意), uAssociatorを利用した関連付け作業, アンケート記入の3段階で行う. 1回の関連付け作業にかかる時間を記録し, それを20回繰り返して行う. 1回の作業は, 机の上にあるuAssociator Cameraを手に取り, 対象物に対してCameraを向け, uAssociatorによる関連付けを成功するまでが含まれる. 実験終了後, アンケートを記入する. このアンケートは, リカート法¹⁴⁾に基づく, 質問内容と評定方法である.

6.2.2 結果

前述した20回の関連付け作業について, 1回の関連付けが成功するまでの平均所要時間を示したグラフを図12に, それぞれの被験者の作業に要した時間の平均値, 最大値, 最小値を示したグラフを図13に示す. また, 被験者に対するアンケートの内容と結果を図14に示す.

6.2.3 考察

結果から, Spot & Snapのユーザビリティに関する各項目について考察する.

学習しやすさ: コンピュータスキルの有無にかかわらず, すべての被験者が簡単な説明だけでuAssociatorを利用できた. 図14のアンケート結果においても, 「使い方を習得するのは簡単だ」の質問に対する13名の平均得点は4.62であり, 非常に学習しやすい印象

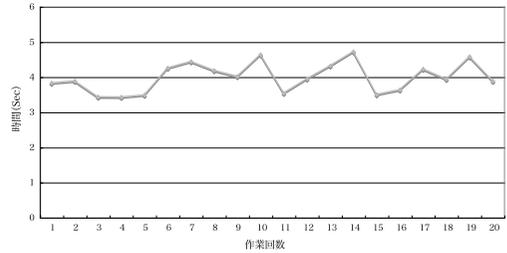


図12 被験者の関連付けに要する作業時間
Fig. 12 Time of mapping operation.

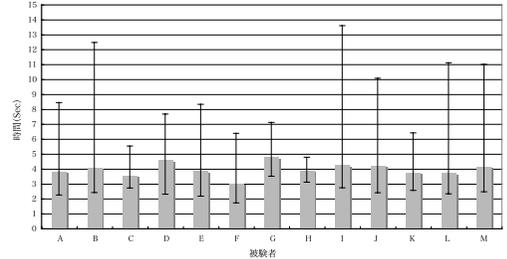


図13 被験者の関連付けに要した作業時間の平均と最大最小値
Fig. 13 Average time of mapping operation.

スポットライトを照射してモノとセンサノードを関連付けることについて	1	2	3	4	5	平均
使い方を習得するのは簡単だ	0	0	0	5	8	4.62
やっけて非常にイライラする	6	4	1	2	0	1.92
やっけて楽しい	1	0	4	2	6	3.92
画像を用いたモノの表示について	1	2	3	4	5	平均
画像を利用してて分かりやすい	0	0	0	2	10	4.77
モノの名前も表示して欲しい	0	0	3	6	4	4.08

図14 アンケート結果
Fig. 14 Result of questionnaire.

を持っていることが分かる.

効率性: 図12より, 被験者は作業回数に関係なく, 作業達成までのインタラクションを平均4秒前後で行えると分かる. 既存のバケットダンプや専門ツールを用いたセンサノードIDの特定作業および手作業による関連性入力作業と比較すれば, 4秒という時間はuAssociatorによる関連付け作業の効率性の高さを示したといえる.

記憶しやすさ: Spot & Snapはスポットライトを照射するという単純な作業しか要求しないため, 1回の説明で被験者は手順を学習できた. よって, 時間をおいても忘れにくいと考えられる.

エラーがあるかまたは起こした際の回復: 図13は, それぞれの被験者は平均4秒程度で作業を行えたことを示しているが, 最大で10秒を超えたケースも見受けられた. これは, 6.1節での実験の際と同様に, 作業時にセンサノードからのバケットが極端に落ちたことが原因であった. このエラーは, 図14の「やっけて非常にイライラする」という質問に対する平均得

点 1.92 に影響を与えていると考えられる。センサノードのパケット落ちに関しては、今後のセンサノードのハードウェアおよびデータ送信プロトコルの信頼性の向上が望まれる。一方で、被験者自身の操作ミスによるエラーは発生しなかった。また、エラーを起こした際にその原因を通知する機構がなかったため、被験者が戸惑う場面が見受けられた。自由記述欄のコメントにも「失敗した際に何が原因なのか表示してほしい」という記述があったため、今後エラー通知機構の実装が求められる。

ユーザの主観的満足度：Spot & Snap の関連付け作業に関するアンケート結果で、設問の平均値は 3.87 であった。平均値を決める際、「やっけていて非常にイライラする」という反転極性を使用した質問はその数値を反転させて計算した。リカートの個人的満足度の中間点は 3.6 であることが知られており、3.87 という数値は、平均よりやや満足度の高いインタフェースであることを示している。ハードウェアおよびソフトウェアの改良により、満足度を今後さらに高めていく必要がある。また画像を用いたモノの表示方法については、図 14 の「画像を利用して分かりやすい」という質問の平均得点が 4.77 であることから、有用であることが分かった。一方、「モノの名前も表示してほしい」という質問に対する平均得点も 4.08 と高く、名前というメタ情報の必要性も分かった。取得した画像よりモノの名前を抽出する機構を実装することにより、よりユーザが利用しやすいシステムが構築できると考えられる。

また、アンケートの自由記述欄では「扱いやすい」「興味深い」等の好意的な意見が目立ったが、特に多かったのは「携帯電話で使えるようにしてほしい」という意見であった。また上述した「失敗したときにその原因を表示してほしい」という意見のほかに「自分のモノの様子が他人から分かれば、犯罪に利用されるかもしれないと少し不安になった」という意見も見受けられた。このような知見を統合して、システム設計、インタフェース設計にフィードバックすることが重要であると考えている。

7. 関連研究

Okadome らの Project Pervasive Association¹⁵⁾ は、センサノードが取り付けられたモノが何であるのかという情報を、蓄積されたセンサデータを解析し、判別しようとする研究である。しかし限られた種類のセンサデータから、そのモノが何であるのかを正確に推測するのは困難である。さらに、モノが何で

あるのかを推測するのに時間を要し、即時にモノをアプリケーションの対象としたい際には適さない。また Beckmann らは、センサノードを家電に取り付ける作業を支援するために、Sensor Installation Kit¹⁶⁾ を提案している。Kit にはセンサノード、家電の一覧がバーコードとともに記載された本と、バーコードを読み取るための小型デバイスが含まれている。Kit ではあらかじめセンサノードと家電との関連付けがされており、小型デバイスで本に記載された家電のバーコードを読み取ると、どのセンサノードを取り付けるべきかの情報がデバイスに表示される。しかしこれは品数が限られた機器だけを対象としており、身の回りにある無数の日用品を対象とすることは困難である。また、ユーザが行うべきインタラクション数も多く、作業に慣れるのに時間を要すると考えられる。

センサノードを取り付けたモノを利用したアプリケーションに関する研究では、1 章で述べた MediaCup¹⁾ や SPECS²⁾ のほか、Decker らの DigiClip⁹⁾、Kiong らの MAX⁸⁾ や Fujinami らの Sentient Artefacts¹⁸⁾ があげられる。しかし、いずれの研究においてもセンサノードとモノの関連付けはプログラマが手動で行うことを想定している。Borriello らの RFTag を利用した忘れ物防止システム¹⁷⁾ では、まずユーザの 1 日のスケジュールを考慮し目的地を検知する。そして、その目的地に持っていくべきモノは何であるのかという情報を、モノにつけられた RFTag と関連付けている。また、仕事先等で新たなモノを即興的に忘れ物の対象にしたい場合、auto-tag という RFTag を取り付けすることで対応している。しかし auto-tag も、ユーザによる関連情報の記述を前提としており作業負荷が高い。

いずれの研究例においても、即興的な関連付けには着目しておらず、また関連付ける情報としてモノの名前だけを対象としている。Spot & Snap は、1 つのインタラクションでユーザの関連付け作業を可能にし、モノの画像を利用した視覚的なインタフェースを有すアプリケーションを実現できる。

8. 今後の展望

今後の展望として、Spot & Snap をユーザの日常生活に適したシステムに改良するためのポイントを考察する。

Spot & Snap は、ユーザがスポットライトを照射したのと同時刻に、照度センサが最高値を示したセンサノード ID を取得している。そのため 6 章で示したように、屋外環境や屋内の窓際周辺等、強い光の下に



図 15 メタ情報入力インタフェース
Fig. 15 Meta data input interface.

あるセンサノードはスポットライト照射による照度センサ値の差を認識できない。照射する前にセンサノードを少し手で覆いかぶせるようにし、スポットライトの照射を検知できるようにしなければならない。Spot & Snap が失敗した原因を特定し、その対処方法を伝達する機構が必要である。LED ライト、カメラ、ディスプレイをすでに持つ携帯電話上での実装を行うことで、特別なデバイスの導入コストをなくし、エラーへの対処方法をディスプレイに表示する機構を有したシステムが実現できると考えられる。また、将来センサノードに赤外線センサが標準で搭載されれば、Spot & Snap 時に赤外線特別なコマンドを送信することにより、センサノード ID の特定が可能になると考えられる。また、電波強度を取得するモジュールをセンサノード上に実装することで、センサノード ID を取得することも考えられる。これらの有効性を検証したうえでセンサノードに対する機能要件を整理し、提案していく必要がある。

一方、多様なアプリケーションを作成するためには、モノの名前等のメタ情報も必要である。モノの名前情報や属性情報は、テキスト入力によるキーワード検索やコンテキスト抽出（たとえば、同じ部屋でもホワイトボードとペンが同時に使用されている際はミーティングをしていると判断）において有効に利用できると考えられる。現在の実装では、名前等のメタ情報が必要なアプリケーションは、メタ情報入力インタフェースを用意することで対応している（図 15）。しかし属性情報の入力は作業の煩わしさにつながるため、自動的なモノのメタ情報抽出機構が必要である。

具体的な方法として、モノに製品名やバーコードが印刷されている場合は、取得した画像からそれらの情報を抽出し、関連付けることができる。また、モノが出荷される際に RFID タグが取り付けられていれば、指向性を持つリーダをカメラに搭載することで、スポットライトの照射時にモノの情報を取得できる。現在、光に反応して ID を送信する RFID システムも研究されており¹⁹⁾、これらのシステムとの連携も図ることができる。類似画像検索技術⁵⁾を応用し、モノの情

報を得る方法も考えられる。将来的にはこれらの技術が組み合わせられた、多様な環境で動作可能なモノのメタ情報取得機能の実現が望まれる。

9. ま と め

我々は、モノを利用したユビキタスサービスにおいて、ユーザの持ち物をユーザ自身がセンサノードを取り付け、サービスの対象として利用する DIY Smart Object Service を提案した。本論文では、DIY Smart Object Service を実現するために、モノとセンサノードの関連付けを容易かつ統一的に行うインタラクション Spot & Snap を提案し、そのインタラクションを実現するためのプロトタイプシステムを試作した。また Spot & Snap を利用したいいくつかのアプリケーションを紹介し、操作実験を行うことによって、4 秒程で関連付けが行えることを確認した。

Spot & Snap によりユーザは、専門知識なしに Smart Object Service を利用することができる。また Spot & Snap の利用を想定することで、アプリケーションの設計が容易となり、アプリケーション開発者の負担が軽減できる。Spot & Snap はセンサノードに対して特別な実装が必要なく、多様なセンサノードでの適用が可能である。これらのことから我々は、Spot & Snap がユビキタスサービスを実社会へ適応する基盤の一助として、役割を担えると考えている。

謝辞 本研究の実施にあたり、有益な助言をいただいた慶應義塾大学徳田高汐研究室諸氏に感謝する。本研究の一部は総務省「ユビキタスネットワーク制御・管理技術の研究開発（ubila プロジェクト）」、および情報推進機構「未踏ソフトウェア創造事業」のもとで実施した。

参 考 文 献

- 1) Beigl, M., Gellersen, H.-W. and Schmidt, A.: MediaCups: Experience with architecture and Use of Computer-Augmented Everyday Objects, *Computer Networks*, Vol.35, No.4, pp.401-409, Elsevier (2001).
- 2) Lamming, M. and Bohm, D.: Specs: Another approach to human context and activity sensing research, using tiny peer-to-peer wireless computers, *The 5th International Conference on Ubiquitous Computing (UbiComp'05)*, Seattle, Washington (2003).
- 3) Holmquist, L.E., Mattern, F., Schiele, B., Alahuhta, P., Beigl, M. and Gellersen, H.W.: Smart-Its Friends: A Technique for Users to Easily Establish Connections between Smart

- Artefacts, *Proc. 1st International Conference on Ubiquitous Computing Demonstration (Ubi-comp'01)*, Atlanta, GA, USA (Sept. 2001).
- 4) Beigl, M., Decker, C., Krohn, A., Riedel, T. and Zimmer, T.: μ Parts: Low Cost Sensor Networks at Scale, *Proc. 7th International Conference on Ubiquitous Computing Demonstration (Ubi-comp'05)*, Tokyo, Japan (2005).
 - 5) Rui, Y., Huang, T.S. and Chang, S.-F.: Image Retrieval: Current Techniques, Promising Directions and Open Issues, *Journal of Visual Communication and Image Representation*, Vol.10, No.4, pp.39–62 (1999).
 - 6) Crossbow Technology - Wireless Sensor Networks, Inertial & Gyro Systems, Smart Dust, Advanced Sensor. <http://www.xbow.com/>
 - 7) Tapia, E.M., Intille, S.S., Lopez, L. and Larson, K.: The Design of a Portable Toolkit of Wireless Sensors for Naturalistic Data Collection, *The 4th International Conference on Pervasive Computing (Pervasive'06)*, Dublin, Ireland, pp.117–134 (2006).
 - 8) Yap, K.K., Srinivasan, V. and Motani, M.: MAX: Human-Centric Search of the Physical World, *The 3rd ACM Conference on Embedded Networked Sensor Systems (SenSys'05)*, San Diego, USA, pp.166–179 (2005).
 - 9) Decker, C., Beigl, M., Eames, A. and Kubach, U.: DigiClip: Activating physical documents, *International Workshop on Smart Appliances and Wearable Computing (IWSAWC'04)*, Tokyo, Japan, pp.388–393 (2004).
 - 10) Warren, C. and Morton, J.: The effects of priming on picture recognition, *British Journal of Psychology*, Vol.23, pp.117–129 (1982).
 - 11) Nakazawa, J., Edwards, W.K., Ramachandran, U. and Tokuda, H.: A Bridging Framework for Universal Interoperability in Pervasive Systems, *The 26th International Conference on Distributed Computing Systems (ICDCS'06)*, Lisboa, Portugal, pp.3–12 (2006).
 - 12) Suzuki, G., Aoki, S., Iwamoto, T., Maruyama, D., Koda, T., Kohtake, N., Takashio, K. and Tokuda, H.: u-Photo: Interacting with Pervasive Services using Digital Still Images, *The 3rd International Conference on Pervasive Computing (Pervasive2005)*, Munich, Germany, pp.190–207 (2005).
 - 13) Nielsen, J. and Levy, J.: Usability Engineering, *Academic Press/AP Professional* (1993).
 - 14) LaLomia, M.J. and Sidowski, J.B.: Measurements of computer satisfaction, literacy and attitudes, *A review. In International Journal of Human-Computer Interaction*, Vol.2, pp.231–253 (1990).
 - 15) Okadome, T., Hattori, T., Hiramatsu, K., Yamada, Y. and Satoh, T.: Project pervasive association: toward acquiring situations with smarter objects, *Proc. Ubicomp'05 Workshop on Smart Object Systems*, Tokyo, Japan, pp.45–52 (2005).
 - 16) Beckmann, C., Consolvo, S. and LaMarca, A.: Some Assembly Required: Supporting End-User Sensor Installation in Domestic Ubiquitous Computing Environment, *The 6th International Conference on Ubiquitous Computing (Ubi-comp'04)*, pp.107–124 (2004).
 - 17) Borriello, G. and Brunette, W.: Reminding about Tagged Objects using Passive RFIDs, *The 6th International Conference on Ubiquitous Computing (Ubi-comp'04)*, Nottingham, England, pp.36–53 (2004).
 - 18) Fujinami, K., Okada, K., Kawsar, F. and Nakajima, T.: Living with Sentient Artefacts, *Proc. 7th International Conference on Ubiquitous Computing Video*, Tokyo, Japan (2005).
 - 19) Ma, H. and Paradiso, J.A.: The FindIT Flashlight: Responsive Tagging Based on Optically Triggered Microprocessor Wakeup, *The 4th International Conference on Ubiquitous Computing (Ubi-comp'02)*, Goteborg, Sweden, pp.160–167 (2002).
 - 20) 象印マホービン株式会社 : みまもりほっとライン—親の元気がポットでわかる .
<http://www.mimamori.net/>

(平成 18 年 6 月 21 日受付)

(平成 18 年 12 月 7 日採録)



米澤 拓郎

2005 年慶應義塾大学環境情報学部卒業。現在、同大学大学院政策・メディア研究科修士課程在学中。無線小型センサノードを用いた分散処理システム、インタラクションシス

テムに関する研究に従事。



榎原 寛

2005 年慶應義塾大学環境情報学部卒業。現在、同大学大学院政策・メディア研究科修士課程在学中。モバイルネットワークにおける帯域統合技術、ユビキタスセンサネットワーク

におけるプライバシーに関する研究に従事。



中澤 仁 (正会員)

現在，慶應義塾大学大学院政策・メディア研究科修士課程助手。主に，分散オブジェクト指向システム，オブジェクト遷移ミドルウェアの研究に従事。



高汐 一紀 (正会員)

1995年慶應義塾大学大学院理工学研究科後期博士課程単位取得退学。電気通信大学電気通信学部情報工学科助手を経て，現在，慶應義塾大学大学院政策・メディア研究科特別研究教員助教授。主に，分散実時間システム，小型デバイス向けモバイルアプリケーション，ユビキタスコンピューティングの研究に従事。日本ソフトウェア科学会，ACM，IEEE 各会員。博士（工学）。



徳田 英幸 (正会員)

1977年慶應義塾大学大学院工学研究科修士。1983年ウオーターラー大学 Ph.D. (Computer Science)。同年カーネギーメロン大学計算機科学科勤務。1990年同学科研究准教授。現在，慶應義塾大学大学院政策・メディア研究科委員長。主に，分散リアルタイムシステム，マルチメディアシステム，超並列・超分散システム，ユビキタスシステムの研究に従事。IEEE，ACM，日本ソフトウェア科学会各会員。