

日常物とセンサノードの関連付け手法の提案

米澤拓郎¹, 榊原寛¹, 中澤仁¹, 永田智大², 高汐一紀^{1,3}, 徳田英幸^{1,3}

¹ 慶應義塾大学大学院 政策・メディア研究科

² NTT ドコモ 総合研究所

³ 慶應義塾大学 環境情報学部

本研究では日常物(以下モノ)とセンサノードの容易な関連付け手法 Spot & Snap を提案する。現在超小型センサノードをモノに取り付け、そのモノの状態を利用したサービスの研究が盛んである。しかし、誰がモノにセンサノードを取り付けるか、そのモノとセンサノードとの関連性は誰が入力するのか、といった問題が存在する。本稿では、ユーザ自身が身の回りのモノにセンサノードの取り付け・関連付けを行い、低コストで自由度の高いサービス利用環境を実現するために、スポットライトとカメラを利用した容易な関連付け手法 Spot & Snap を提案する。また Spot & Snap のプロトタイプを実装し、Spot & Snap を利用した多様なアプリケーション例を示すことでその有用性を議論する。更にその操作実験を通じ、平均4秒以内の即興的な関連付けが行えることを示す。

A Technique for Associating Everyday Objects and Sensor Nodes

Takuro Yonezawa¹, Hiroshi Sakakibara¹, Jin Nakazawa¹,
Tomohiro Nagata², Kazunori Takashio^{1,3}, Hideyuki Tokuda^{1,3}

¹ Graduate School of Media and Governance, Keio University

² Research Laboratories, NTT DoCoMo, Inc.

³ Faculty of Environmental Information, Keio University

This paper introduces Spot & Snap interaction technique that enables end-users to associate sensor nodes to everyday-life objects easily. By attaching sensor nodes to everyday-life objects, are what we call smart objects, users can augment the objects digitally. When using smart object services, a semantic connection between sensor node and 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, non-expert 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

Mark Weiser は、ユビキタスコンピューティングを「多数のコンピュータを物理的環境を介して利用できるようにし、なおかつそれらがユーザーには見えないように工夫しつつ、コンピュータ能力の活用を増進する方法」と定義した [1]。ユビキタスコンピューティングを実現するための要素技術として、オフィスや家庭の様々な道具や場所に埋め込める超小型無線センサノードの重要性は高く、多数の研究開発が行われている [2, 3, 4, 5]。本稿ではセンサノードを、照度・温度・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データを処理し、人や環境の状態を抽出することによって、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、センサノードを日常物(以下モノ)に取り付け、モノやモノ周辺の状況を利用したサービス(以下スマートオブジェクトサービス)の研究が活発である。例えばセンサノードをコップに取り付け、飲み物の温度を感知し冷める前にユーザに通知する MediaCup [6] や、センサノードを身の周りのものに取り付け、忘れ物があればユーザに通知する SPECS [7]、また取り物の発見支援サービスを提供する MAX [8] 等が挙げられる。我々の生活はモノとのインタラクションに溢れており、ユビキタスコンピューティング環境を実現するために、スマートオブジェクトサービスの必要性は今後より一層高まっていくと予想される。

スマートオブジェクトサービスの実現には、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、という実世界での関連性を情報空間で反映する必要がある。本稿では、関係性を情報空間に反映させる作業を、関連付けと定義する。関連付けは、センサノードを区別するセンサノード ID と、モノを示す情報とを結びつけることで行う。よって関連付けは、「センサノード ID の特定」と「センサノード ID とモノの関連性の入力」の2つの作業から成る。この作業をユーザが行う際の問題点は、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。そのため専用のツールを使うか、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元に 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つを特定するのは更に困難となる。
- **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、コンピュータ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。先に述べた先行研究では、この関連付けを簡単に行う方法については焦点を当てておらず、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。もしくは、予めセンサノードを製品の工場出荷時に取り付け必要な設定を行っておくことで、ユーザ自身が関連付けを行う必要はないと仮定している。しかし専門家を呼んだり製品として購入したりすることは、いずれもユー

¹ 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)」及びハイテク・リサーチ・センター整備事業「e-ケア型社会システム形成」の下行われた。

ずかに金銭的・時間的負担を与えることになる。

本研究の目的は、モノとセンサノードの関連付けを容易に行う手法をユーザに提供し、ユーザ自身が自由に様々なサービスを利用・構築できる環境を実現することである。ユーザ自身が関連づけを行い実現するスマートオブジェクトサービスを、本稿ではDIYスマートオブジェクトサービスと呼ぶ。DIYスマートオブジェクトサービスは、予め設定されたスマートオブジェクトの利用や専門家による設定は必要なく、ユーザに専門的知識を要求することもない。ユーザが自分の持ち物にセンサノードを取り付け、関連付けを行うことで実現されるサービスである。我々は目的を達成するために、センサノードとモノの関連付け手法 Spot & Snap を提案する。Spot & Snap では、スポットライトが取り付けられたカメラを利用して関連付けを行う。関連付けの作業は、関連付けの対象物（センサノードを取り付けたモノ）を照射しながら撮影するだけである。「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つく」ということを意識する以外、ユーザにセンサノードID等の専門知識は一切要求しない。

本稿はまず Spot & Snap の詳細について述べ、その後に Spot & Snap を実現するシステムを説明する。更に Spot & Snap を利用した実現アプリケーション例を述べ、実装システムの評価を行う。そして関連研究と今後の課題を述べ、最後に本稿のまとめを記す。

2 Spot & Snap

本稿では、問題点で述べた関連付け作業を容易にし、DIYスマートオブジェクトサービスを実現するインタラクション Spot & Snap を提案する。Spot & Snap によるスマートオブジェクトサービスの利用プロセスを図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する。そして (B) 自分の持ち物にセンサノードを取り付ける。(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スマートオブジェクトサービスを利用できる。次節で、Spot & Snap の動作の詳細を述べる。

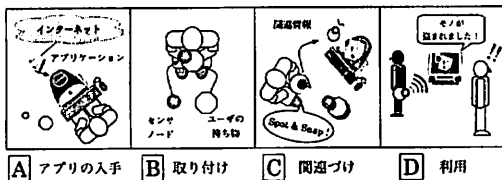


図1: Spot & Snap に基づくサービス利用プロセス

2.1 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、及びセンサノードIDとモノの関連性入力、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2-3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、スマートオブジェクトサービスを実

現するアプリケーションに登録する。画像と関連付ける意味は、3.3節に後述する。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とモノを関連付けることを可能にする。



図2: Spot & Snap における関連付け動作

2.2 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において、Spot Snap は一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、自身のIDとセンサ値を含んだパケットを無線でシンクノードに送信し、各コンピュータはそのデータをシンクノードを経由して受け取ることができるものとする。

スポットライトのスイッチをONにした時間を T_s とした時、 $T_s < t_s < T_s + N$ 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 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 とした時、 $T_e < t_e < T_e + N$ 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。この時、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（例えば図3の場合、 S_1 がユーザによってスポットライトを当てられたセンサノードと判断する）。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 及び 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 落ちを考慮し、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 秒とした時 $N = 2i$ とした。このアルゴリズムを用いた時のセンサノードID特定成功率は、5章で述べる。

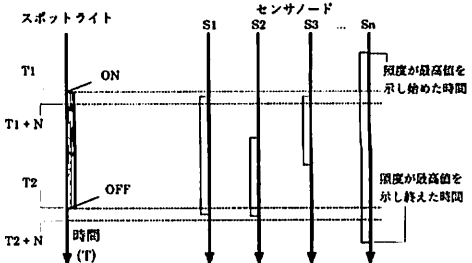


図3: センサノードID特定の例

2.3 モノの情報の表現方法

センサノードIDと関連付けるモノの情報を情報空間でどう表現するべきであるかは、議論を要する。名前、属性、特徴等、様々なモノの表現方法が考えられる。大きく分けて、(1) 人が自然に理解でき、(2) 多様なアプリケーションで

利用可能な表現方法であること、の2点を満たすことが理想的であると考えられる。

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する。一般的には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つけ、モノの名前を認識し、性質を理解する [9]。「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。「コップ」がどんなコップなのか、色や形をテキストとして表現することも可能であるが、多数のモノの関連付けを行うことを想定すれば、現実的ではない。このため、ユーザにとってモノの表現手法として画像を利用することは実用的かつ有効であると考えられる。

一方、モノの名前や性質というメタ情報での表現も、多様なアプリケーションを構築する上では重要だと考えられる。現在までに、多くの研究者がモノのメタ情報を自動抽出する手法の構築を試みている。例えばコンピュータビジョンの研究では、モノの画像からメタ情報を抽出する多数の手法が提案されている。しかしながらこれらの技術は、ある特定の条件下では有効に動作するが、様々な環境下で多くのモノを対象として動作する一般的な手法としては未だ確立されていない。

Spot & Snap ではモノのメタ情報は利用しないため、情報空間でのモノの表現方法としては制限が存在する。しかし本稿では、メタ情報を利用しないという欠点よりも、モノの画像を用いた時の利点に焦点を当て、本稿では、モノの画像情報だけで実現できるアプリケーション例を4章で示す。そして画像だけでも様々なアプリケーションを開発するために十分な情報となりうることを述べる。その上で今後の展望として、モノのメタ情報取得手法を、8章で議論する。

2.4 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予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一切必要ないため、導入も容易である。照度センサが備わったセンサノードは全て Spot & Snap の対象とできる。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5章に示す。

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。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、ドライヤー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、ドライヤーの小型化が難しく、カメラと一体化した小型装置にはなりえない。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5章に示す。

センサノードを特定する手法は、他にも考えられる。例えば、電波強度を利用したり、単純にバーコードをセンサノードに取り付けてそれを読み取る方法等が挙げられる。7章でこれらの手法を挙げ、その際に発生する問題点を示す。

● スポットライトによる直感的インタフェース

一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、視覚的に分かり易い直感的操作を可能にする。また Spot & Snap を実現するためのデバイスは、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タルカメラを利用して実現できると考えられる。

3 実装システム: uAssociator

uAssociator は、Spot & Snap を実現するための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 で実装した。

3.1 ハードウェア

uAssociator で利用するデバイスとして、スポットライトと WEBカメラ、スポットライトの ON / OFF を検知する機構から構成される uAssociator Camera を試作した。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ドイツ TecO 研究所で開発された μ Part [3] と、CrossBow 社が販売している MICA DOT [4] 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。



図 4: uAssociator Camera を用いた関連付けの様子

3.2 ソフトウェア

本節ではまず uAssociator のソフトウェアをまず述べ、その後アプリケーションが関連付け情報を利用するための API を説明する。

3.2.1 構成

uAssociator のソフトウェア構成図を図5に示す。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。センサデータ抽象化部では、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 [10]。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。センサデータ ID 識別部は、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、2.2 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かと判断したセンサノード ID を関連付け部に渡す。画像取得部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、関連付け部に渡される。関連付け部は受け取ったセンサノード ID と画像を関連付け、uAssociator Image を生成する。uAssociator Image はアウトプット部に渡される。アウトプット部は、コネクション部が管理するアプリケーションリストを参照し、アプリケーションに uAssociator Image を送信する。アプリケーションリストには、uAssociator Image を要求するアプリケーションの IP アドレス、ポート番号等の情報が保存されている。またアウトプット部は、後の利用のためにローカルファイルシステム内 uAssociator Image を保存する。アプリケーションは uAssociator Image を利用することでスマートオブジェクトサービスを実現できる。uAssociator Image は、JPEG 形式の画像ファイルであり、そのコメント領域にセンサノード ID 等の関連付け情報を XML 形式で保存する (図 6 参照)。

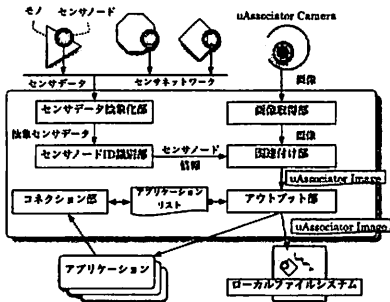


図 5: uAssociator のソフトウェア構成図

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE association_info SYSTEM "uAssociator.dtd">
<association_info>
  <sensor_info>
    <type> uPart </type>
    <id> 1.2.3.4.0.1.0.12 </id>
  </sensor_info>
  <object_info>
    <name> Unnamed Object </name>
    <attributes></attributes>
  </object_info>
  </timestamp> 1163488439679 </timestamp>
</association_info>
```

図 6: uAssociator Image に保存されるメタデータの例

3.2.2 アプリケーションとの連携

uAssociator は、アプリケーションが uAssociator Image を 2通りの方法で扱えるように、複数の API を提供している。1つ目は、uAssociator に直接接続し、uAssociator Image を取得する直接利用のための API である。直接利用は、アプリケーション実行時に新たなスマートオブジェクトをサービスの対象としたい場合に有効である。直接利用の API を使用したプログラム例を図 7 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。アプリケーションは SmartObject クラスで保持するセンサノードの種類やセンサノード ID を参考として、スマートオブジェクトサービスを構築できる。

2つ目は、ローカルファイルシステム内に保存された uAssociator Image ファイルを読み込んで利用する。間接利用のための API である。間接利用の機能を提供することで、一

度関連付けたスマートオブジェクトを後で別のアプリケーションに登録したい際に、再び Spot & Snap を行うことなく利用できる。間接利用は、ユーザは GUI から uAssociator Image ファイルを開くか、ファイルを Drag & Drop することでアプリケーションに読み込ませることで実現される (図 8)。Drag & Drop 機能を実装したプログラム例を図 9 に示す。UAssociatorImageReader クラスは、uAssociator Image ファイル内に記述された XML を解析し、SmartObject クラスのオブジェクトを生成する。

```
public class ExApp implements AssociationListener{
  public void spotAndSnap(SmartObjectEvent e){
    sObjectList.addElement(s.getSmartObject());
  }
  public void processing(){
    if( isEventOccured((SmartObject) sObject)
      Light.on();
    }
  }
}
```

図 7: 直接利用のための API を利用したプログラム例

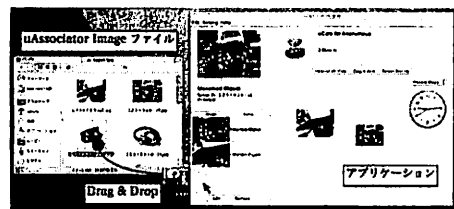


図 8: Drag & Drop によるモノの登録

```
public class ExApp2 implements DropTargetListener{
  public void drop(DropTargetDropEvent e){
    ....
    UAssociatorImageReader reader =
      new UAssociatorImageReader Reader();
    SmartObject sObject =
      reader.getSmartObject(file.getName());
  }
}
```

図 9: 間接利用のための API を利用したプログラム例

4 アプリケーション

Spot & Snap の有効性を議論するため、いくつかのアプリケーションを実装した。以下にその詳細について述べる。

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 10)。例えば、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、またリアルタイムにモノの状態を可視化するので (例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。

uAlert

uAlert は、引き出しやタンスの中に閉ま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。

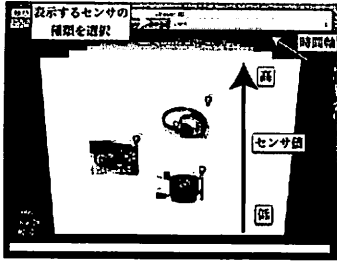


図 10: uViewer のプロトタイプ

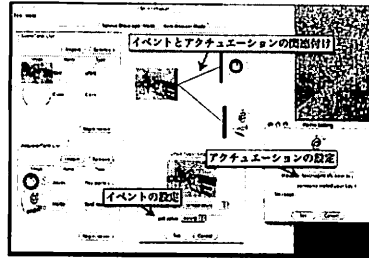


図 11: uServiceMaker のプロトタイプ

uCare

uCare は、一人暮らしの高齢者の様子を家族が見守れ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モニタリングする。uCare はそれ自身が簡易 WEB サーバの役割を果たし、家族や介護者は高齢者がどのモノを使っているか携帯電話等のブラウザを利用して把握することができる。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 を通じて認識でき、無事を確認することができる。類似したサービスに象印社の iPot[11] が挙げられる。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。

uReminder

uReminder は、ユーザが家を出る際に忘れ物があれば知らせてくれるアプリケーションである。uReminder は 2 種類の異なる性質を持つモノを登録し、それらの動きを監視することで忘れ物があることを検知する。1 つめはユーザがかけられる際にいつも装着する種類のモノ（例：靴）であり、2 つめは忘れる可能性がある種類のモノ（例：財布や鍵）である。uReminder は、もし登録された財布や本など忘れる可能性のあるモノが動いていないのに靴だけが動き出すと、ユーザが忘れ物をして外に出ようとしていると判断し、アラームを鳴らしたり忘れ物の画像を表示したりしてユーザに通知する。ユーザは、Spot & Snap と uReminder の GUI を通じ、それぞれの種類のモノを登録する。

uServiceMaker

uServiceMaker は、ユーザが簡単に独自のスマートオブジェクト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スマートオブジェクト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。例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。

議論

これらのアプリケーションから、Spot & Snap の有効性と、メタ情報の必要性について議論する。

- **Spot & Snap の有効性:** 上述したサービスは、どれも普段利用している持ち物をターゲットとしている。例えば

uCare で高齢者の生活の様子を知るためには、普段高齢者が利用するモノをモニタリングする必要がある。Spot & Snap が提供する関連付け手法は、ユーザの持ち物をそのまま利用できるアプリケーションの実現につながるため、有効であると考えられる。

- **メタ情報の必要性:** 関連付けにモノのメタ情報（名前や属性）を利用すれば、上述したアプリケーションにとって様々な利益を提供できる可能性がある。例えば、モノの属性を認識できればユーザはわざわざモノの種類を気にせず、uReminder にモノを登録できるという機能が付与できる。その一方で、これらのアプリケーションを実現するには、必ずしもモノのメタ情報が必要となる訳ではないことが分かる。なぜならこれらのアプリケーションは、ユーザがモノの画像を通じてモノを認識することで、サービスの要件を満たしているからである。これらのことから我々は、Spot & Snap がメタ情報を取得できないという制限が必ずしも欠点につながるとは考えていない。

5 評価

本章では、本稿で提案した Spot & Snap 及び実装システム 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 の使用実験を行った。またユーザビリティの評価として、複数の被験者で uAssociator の利用実験を行った。

5.1 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 Part と DOT 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（部屋の全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づけ作業を 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を判定する。またそれぞれの場所には関連づ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 1 に示す。

表 1: 場所毎のセンサノード ID 特定の成功率

	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%

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パ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 の照度センサが常に最高値を示したためである。一方、窓際で

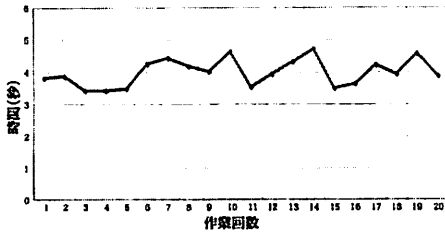


図 12: 関連付けに要した平均作業時間

は μ Part、DOT いずれも照度が最高値となってしまう、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たり、手で影を作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る。

5.2 ユーザビリティ評価

ユーザビリティの指標 [12] を表す学習しやすさ、効率性、記憶しやすさ、エラーがあるかまたは起こした際の回復、ユーザの主観的満足度の 5 つに関して、実験を行い評価する。

5.2.1 実験

実験の対象者は、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、部屋の全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 [13] に基づく、質問内容と評定方法である。アンケートの内容と結果は次節に示す。

5.2.2 結果と考察

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結果を表 14 に示す。以上の結果から、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関わらず、全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」という質問に対する 13 人の平均得点は 4.62 であり、非常に学習しやすい印象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。

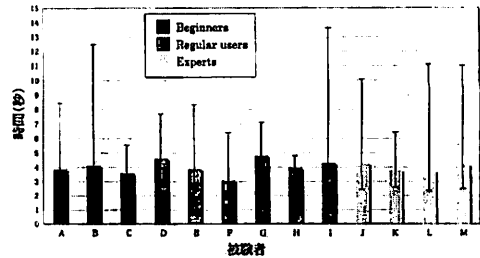


図 13: 関連付けに要した作業時間の平均と最大最小値

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
やっけて嬉しい	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
画像を利用して分かり易い	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

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 の特定作業及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言える。

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、一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づらと考えられる。実際、1ヶ月後に複数の被験者に手順を覚えていたか尋ねた結果、全員がはっきりと覚えていた。

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。センサノードのバ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及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欲しい」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にて、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、平均より満足度の高いインタフェースであることを示している。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易い」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欲しい」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。今後モノの名前を抽出する機構を実現することにより、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欲しい」という意見であった。また「自分のモノの標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、今後のシステムにフィードバックすることが重要であると考えている。

6 関連研究

関連研究として、センサノードとモノの関連付け手法、センサノード ID 特定手法の2つに分け、説明する。

6.1 センサノードとモノの関連付け手法

センサノードとモノの関連付けに関する研究として、センサノードをモノに取り付ける前での関連付けと、後での関連付けの2通りでの研究例が挙げられる。

センサノードをモノに取り付ける前での関連付けとして、Sensor Installation Kit[14]が挙げられる。Sensor Installation Kitは、ユーザが家庭に Home Energy Tutor というアプリケーションを導入する際に必要となる、センサノードの機器への取り付け作業を支援する。Kitにはどのセンサノードをどの機器のどの場所に取り付けるかという関連情報が予め定義されており、ユーザはその情報を参照しながら機器にセンサノードを正確に取り付けることができる。しかし予め定義しておくということは、限られたセンサノードを限られた機器にしか取り付けられないという欠点につながり、家庭の多数のモノを対象とした関連付け手法としては適さない。

またセンサノードを取り付けた後の関連付けとして、Project Pervasive Association[15]が挙げられる。Pervasive Associationでは、まずモノに取り付けられたセンサノードの値の変化からモノの使われ方(引く、回す、押す等)を特定する。そして、モノの使われ方と名前が関連づけられた辞書を参照することで、モノの名前を類推している。モノの名前というメタ情報を利用するアプリケーションにとっては、有益な情報を提供可能であると考えられる。しかし、この手法には以下の三つの問題があると考えられる。第一は、即興的には関連付けが行われない点である。モノの使われ方を認識するために、そのモノが一定回数使用されなければならない。第二に、必ずしも全てのモノの名前がその使われ方によって排他的に定義できるとは限らないため、モノが一意に判別できない可能性がある。第三に、名前が一意に判別できたとしてもそれは一般名詞であるため、ユーザに対しモノの個体情報の提供が必要なアプリケーションには不適切である。Spot & Snapは即興的な関連付けを実現すると共に、モノの画像を利用しているためユーザはモノの個体識別ができるという利点がある。

6.2 センサノード ID 特定手法

デバイス識別に利用される技術を挙げ、それぞれがセンサノード ID 特定に適しているかどうかを議論する。識別手法の最も単純な例として、センサノード ID が記述されたバーコードやビジュアルマーカをセンサノードに貼り付け、画像認識を利用して ID を識別する手法が考えられる。しかしバーコードは取り付けるための物理的スペースが必要となり、センサノードの小型化を阻むという欠点がある。また、赤外線センサをセンサノードに搭載し、通信することでセンサノード ID を認識する手法も考えられる。しかし赤外線は他のセンサに比べ、センサノードに取り付けられる頻度が低い。センサノード識別のためだけに赤外線センサをセンサノードに取り付けることは、コスト高につながるという欠点がある。またデバイス間の関連付けの過程で行うデバイス認識手法でも、それぞれセンサノード特定に適用するには問題を含んでいる。デバイス間の関連付け手法として、デバイス同士の

電波強度を利用した手法[16]、デバイス間の協調動作(加速度センサ値)を利用した手法[17, 18]、また電波強度と協調動作を両方利用した手法[19]等が挙げられる。電波強度を利用するためには、同一の規格を持ったセンサノードが必要となってしまうため汎用性に欠けてしまったり、近い場所に別のセンサノードがあると間違えて他の ID を読み取ってしまう恐れがあるという欠点がある。一方で協調動作を利用した手法はセンサノード識別には有効であるが、その後のモノの情報取得の際に問題が発生する。本稿で提案したようにモノの画像を撮影するという手法を適用した際に、協調動作を起こして・撮影する、という2段階のインタラクションが必要であるからだ。

一方、Spot & Snapと同様に光を利用した識別手法として FindIT Flashlight[20]や RFIG[21]が挙げられる。これらの手法では、あるパターンを含んだ光を照射することで、光センサが取り付けられた RFID が ID (FindIT Flashlight の場合)あるいは ID + 位置情報(RFIG の場合)を特定できる。これらの手法は光を利用しているということで本研究のアプローチと非常に似ている。将来的に RFID とセンサノードが融合した際に、Spot & Snapにおいてもこれらの手法と融合して動作できると考えられるが、現時点ではこれらの手法はセンサノード側に特別な実装を必要としている。一方 Spot & Snap ではセンサノード側に特別な実装を要求していないため、多様なセンサノードを対象として容易に実現可能であるという利点がある。

7 今後の展望

本節では今後の展望について述べる。今後の課題として、センサノード識別精度の向上、メタ情報の取得、携帯電話を利用した Spot & Snap の実装の3つを挙げる。

7.1 センサノード ID の識別精度の向上

センサノード ID の識別精度を向上する方法として、センサノードにおける時刻情報の利用が挙げられる。現在のセンサノード ID 識別アルゴリズムでは、単純に uAssociator がセンサネットワーク内のシンクノードからセンサデータを受け取った時刻を利用している。これは、本研究で対象としたセンサネットワークが、1ホップでシンクノードヘデータを送信するという単純な構成となっており、データ送信の遅延がほとんど無視できるからである。しかし将来、マルチホップを利用した複雑な構成のセンサネットワークを対象とした際には、遅延が問題となってセンサノード ID の特定精度が低下することが予想される。センサノードが時刻情報を扱えれば、センサノード上でスポットライトに照射された時間を正確に検知でき、センサノード識別の精度を向上できると考えられる。そのためには、センサネットワーク内の時刻を正確に同期する必要があり、RBS アルゴリズム[22]等の時刻同期手法が利用できると思われる。

7.2 メタ情報の取得

本稿ではモノの画像だけで様々なアプリケーションが実現できることを述べた。一方メタ情報の利便性もやはり高いと考えられるため、本節ではその取得方法について議論する。モノのメタ情報を取得する方法として先に述べた Pervasive Association の手法以外に、モノに取り付けられた識別子の利用、画像認識利用の2つが挙げられる。

- **モノに取り付けられた識別子の利用:** モノにバーコードや RFID 等が取り付けられていれば、Spot & Snap 時にそれらを読み取ることでモノのメタ情報が取得できる。しかし、必ずしも全てのモノがこれらの識別子を有しているとは限らないため、この手法では限界がある。

- **画像認識技術の利用:** 一方、画像認識技術を利用することでモノの識別子メタ情報が取得できると考えられる。例えばモノの表面に書かれている文字（製品名や特徴）をOCR技術を利用して読み取り、メタ情報として利用できる。しかし文字が書かれていないモノでは適用できないという制限がある。一方で、類似画像検索技術を利用したモノのメタ情報抽出方法も考えられる。CBIRシステム[23, 24]は、モノの画像から色や形・テクスチャの特徴を抽出し、データベース内の画像と照合して類似した画像を検索できる技術である。モノの画像とメタデータが関連づけられたデータベースに対してCBIRを適用することで、Spot & Snapで撮影したモノの画像のメタ情報を取得できると考えられる。しかし家庭内の多数のモノを全て識別するためにはメタ情報が埋め込まれた大量の画像を保持するデータベースが必要となるという問題がある。また現在のCBIRシステム自体には、特定の条件下でしか画像認識をうまく行えないという問題も存在する。

以上、メタ情報の取得方法を述べたが、モノの情報を取得するための最良な方法はまだ確立されていない。現実的な方法として、これらオブジェクト認識の技術を組み合わせた、ベストエフォート型のメタ情報取得システムを構築する必要があると考えられる。

7.3 携帯電話を利用した Spot & Snap の実装

現在多くのユーザが携帯電話を保有しているため、携帯電話で Spot & Snap を実現することでより容易に家庭にサービスをインストールできると考えられる。Spot & Snap はライトとカメラというシンプルなデバイスで実現できるため、それらを有する携帯電話での実装は現実的であると考えられる。更に携帯電話のディスプレイや音・振動等の機能を利用することで、より直感的なインタフェースを実現することができると考えられる。例えばもしセンサノードが強い光の下にいるせいで、センサノードがセンサノードの照射の下にいるせいで、センサノードがセンサノードを手で隠すという手順をユーザに知らせることができると考えられる。

8 まとめ

モノとセンサノードの関連づけは現実空間と情報空間を融合し、多様なサービスを実現するための最初のステップである。本稿ではその重要さに注目し、ユーザ自身がセンサノードをモノに取り付け、関連付けを容易かつ直感的に行えるインタラクション Spot & Snap を提案した。Spot & Snap を利用することで、ユーザ自身が普段利用する持ち物を利用する自由度の高いスマートオブジェクトサービスを実現できる。本稿では Spot & Snap を実現するためのプロトタイプシステムを試作し、実際に Spot & Snap を利用したアプリケーションを実装した。また操作実験を行うことによって、その有効性を評価した。更に今後の展望として、システムの改良点やモノのメタ情報取得方法を考察した。

Spot & Snap よりユーザは、専門知識なしにスマートオブジェクト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、プログラマ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。

参考文献

- [1] Mark Weiser. Some computer science issues in ubiquitous computing. *Commun. ACM*, 36(7):75-84, 1993.
- [2] Lars Erik Holmquist, Friedemann Mattern, Bernd Schiele, Peter Alahutla, Michael Beigl, and Hans-W. Gellersen. Smart-its

friends: A technique for users to easily establish connections between smart artefacts. In *International Conference on Ubiquitous Computing (UbiComp)*, 2001.

- [3] Michael Beigl, Christian Decker, Albert Krohn, Till Riedel, and Tobias Zimmer. *It parts: Low cost sensor networks at scale*. In *UbiComp Demo Session*, 2005.
- [4] Crossbow technology inc. <http://www.xbow.com/>.
- [5] Emmanuel Manquie Tapia, Stephen S. Intille, Louis Lopez, and Kent Larson. The design of a portable kit of wireless sensors for naturalistic data collection. In Kenneth P. Fishkin, Bernd Schiele, Paddy Nixon, and Aaron J. Quigley, editors, *Persuasive*, volume 3968 of *Lecture Notes in Computer Science*, pages 117-134. Springer, 2006.
- [6] Michael Beigl, Hans-W. Gellersen, and Albrecht Schmidt. Mediacups: experience with design and use of computer-augmented everyday artifacts. *Computer Networks (Amsterdam, Netherlands: 1999)*, 35(4):401-409, 2001.
- [7] Milk Lamming and Denis Bohm. Specs: Another approach to human context and activity sensing research, using tiny peer-to-peer wireless computers. In *International Conference on Ubiquitous Computing (UbiComp)*, 2003.
- [8] Kok-Kiong Yap, Vikram Srinivasan, and Mehl Motani. Max: human-centric search of the physical world. In *SensSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 166-179, New York, NY, USA, 2005. ACM Press.
- [9] C. Warren and J. Morton. The effects of priming on picture recognition. In *British Journal of Psychology*, volume 23, pages 117-129, 1982.
- [10] Jin Nakazawa, Hideyuki Tokuda, W. Keith Edwards, and Umakishore Ramachandran. A bridging framework for universal interoperability in pervasive systems. *icdcs*, 0:3, 2006.
- [11] ipot. <http://www.mlmamori.net/>.
- [12] Jakob Nielsen. *Usability Engineering*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1995.
- [13] Mary J. LaLomia and Joseph B. Sidowski. Measurements of computer satisfaction, literacy, and aptitudes: A review. *International Journal of Human-Computer Interaction*, 2, 1990.
- [14] Chris Beckmann, Sunny Consolvo, and Anthony LaMarca. Some assembly required: Supporting end-user sensor installation in domestic ubiquitous computing environment. In *The 6th International Conference on Ubiquitous Computing*, pages 107-124, 2004.
- [15] Takeshi Okadome, Takashi Hattori, Kaoru Hiramatsu, and Yutaka Yanagisawa. Project pervasive association: Toward acquiring situations in sensor networked environments. *International Journal of Computer Science and Network Security*, 6(3B), 2006.
- [16] Waylon Brunette, Carl Hartung, Ben Nordstrom, and Gaetano Borriello. Proximity interactions between wireless sensors and their application. In *WSNA '09: Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, pages 30-37, New York, NY, USA, 2003. ACM Press.
- [17] Jun Rekimoto. Synctap: synchronous user operation for spontaneous network connection. *Personal Ubiquitous Comput.*, 8(2):126-134, 2004.
- [18] Ken Hinckley. Synchronous gestures for multiple persons and computers. In *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology*, pages 149-158, New York, NY, USA, 2003. ACM Press.
- [19] Hans Gellersen, Gerd Kortuem, Albrecht Schmidt, and Michael Beigl. Physical prototyping with smart-its. *IEEE Pervasive Computing*, 3(3):74-82, 2004.
- [20] Hongshen Ma and Joseph A. Paradiso. The findit flashlight: Responsive tagging based on optically triggered microprocessor wakeup. In *UbiComp '02: Proceedings of the 4th international conference on Ubiquitous Computing*, pages 160-167, London, UK, 2002. Springer-Verlag.
- [21] Ramesh Raskar, Paul Beardsley, Jeroen van Baar, Yao Wang, Paul Dietz, Johnny Lee, Darren Leigh, and Thomas Willwacher. Rflg lamps: interacting with a self-describing world via photosensing wireless tags and projectors. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 406-415, New York, NY, USA, 2004. ACM Press.
- [22] Jeremy Elson, Lewis Girod, and Deborah Estrin. Fine-grained network time synchronization using reference broadcasts. *SIGOPS Oper. Syst. Rev.*, 36(SI):147-163, 2002.
- [23] Y. Rui, T. Huang, and S. Chang. Image retrieval: current techniques, promising directions and open issues, apr 1999.
- [24] Sameer Antani, Rangachar Kasturi, and Ramesh Jain. A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video. *Pattern Recognition*, 35(4):945-965, April 2002.