

u-Photo: ユビキタス情報を付加した画像を実現する 環境情報スナップショットの開発

鈴木 源 太[†] 岩 本 健 嗣[†]
高 汐 一 紀[†] 徳 田 英 幸^{†, ††}

写真撮影によって、写っている情報家電やセンサの情報を取得する環境情報スナップショットを開発する。撮影したデジタル写真は、情報家電やセンサ上のアプリケーションを起動する視覚的なコンソールとなる。ホームネットワーク環境においては数多くの情報家電やセンサが配置されると予想されるが、それら実際の機器と、それらのネットワーク上の情報をどのように結びつけて取得するかが課題である。我々の提案する u-Photo では、情報家電やセンシングエリアを写真に取ることで、それらのネットワーク上のアプリケーション情報をデジタル写真上に GUI として付加し、そこからすぐに情報取得や、アプリケーション起動が実現できる。加えて、ビデオの再生中といった動作中のアプリケーションの状態も写真の視覚情報とともに取得し、u-Photo を通じて他の情報家電への同じ作業の再現といった複数機器間の協調動作も実現できる。

u-Photo: Ubiquitous Environmental Snapshot Capturing Contextual Information in Home Network Environment

GENTA SUZUKI[†], TAKESHI IWAMOTO[†], KAZUNORI TAKASHIO[†]
and HIDEYUKI TOKUDA^{†, ††}

In this paper, we describe a novel method named u-Photo for capturing contextual information of networked appliances and sensors in ubiquitous home network environment. In ubiquitous home network environment, it is difficult for users to understand the linkage of the physical and virtual world. In u-Photo, users can get intuitively embedded contextual information by taking photographs and start distributed application from the photograph without any configuration of destination such as IP address. In addition, u-Photo also records status of application to realize multi-device collaborative applications such as suspend/resume of user's task.

1. はじめに

近年、コンピュータの小型化とその遍在化により、「いつでもどこでも」ユーザがコンピュータを利用できるユビキタスコンピューティング環境が実現しつつある。本環境では、さまざまな機器やものが計算能力とネットワークコネクティビティを備え、それらがネットワークを介して協調動作を行うことができる。

本環境の実現は、屋内、屋外双方でユーザに利益をもたらすことができるが、特に屋内環境はホームネットワーク環境とも呼ばれ、インフラ整備が進んでいる。たとえば、ネットワークコネクティビティを持つ家電機器である情報家電は既にその一部が市販されており、またそれらを協調動作させるためのミドルウェアである Jini¹⁾ や HAVi²⁾、UPnP³⁾ などが提案されている。また、ネットワークコネクティビティを持つセンサが情報家電と連携したり⁶⁾、屋内環境モニタリングを実現する試みもされている。

ユビキタスコンピューティング環境には、ユーザがコンピュータを意識することなく利用するという要件がある。実際にコンピュータやセンサを部屋の壁や床に埋め込み、見た目は通常の部屋と変わらないユビキタス空間⁴⁾が提案されている。

確かに、埋め込まれたコンピュータやセンサは、それらが提供するアプリケーションや環境情報といった仮想空間上の情報が重要であり、ハードウェアの物理空間上での視覚情報は必要ない。しかし、「この環境情報は、部屋のどの場所のものなのか」というように、物理空間上に照会して認識すべき仮想空間上の情報もある。また、「この情報家電を操作するアプリケーションはどのように取得すればよいのか」というように、視覚情報からものや場所に関する仮想空間上の情報を取得する要求もあるだろう。

そこで我々は、環境側のセンサ情報や、情報家電機器上で動作するアプリケーション情報といった仮想空間上の見えない情報を、写真撮影という行為によって取得し、写真上に表示することで、物理空間の視覚情報の上に仮想空間の情報を重ねる新たな手法を提案する。本稿で提案するデジタル写真メディアは u-Photo (ubiquitous-Photo) と呼ばれ、u-Photo を閲覧、生成するためのサブシステムは u-Photo Viewer および u-Photo Creator と呼ばれ

[†] 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

る．これらにより，ユーザは仮想空間上の情報をより直感的に認識できる．また本提案におけるデジタル写真メディアは，情報表示のための利用だけでなく，アプリケーション起動のためのユーザインタフェースとしても利用することができる．加えて，アプリケーションの撮影時の動作状態などを記録するため，情報の保持や携帯が可能となり，リアルタイムでの利用以外の二次利用も実現できる．

本稿は全8節によって構成される．第2節では，本研究が対象とするホームネットワーク環境についてまとめ，本研究の取り組むべき課題を明らかにする．次に第3節で，課題を解決する手法としてデジタル写真メディア利用したアプローチについて説明する．第4節では，u-Photo，u-Photo Viewer および u-Photo Creator のについて説明し，第5節でプロトタイプ実装について述べる．第6節にて，基本性能を定量的に評価し，第7節で本研究と関連研究と比較する．最後に，第8節でまとめと今後の課題について言及する．

2. ホームネットワーク環境

本研究において前提となるホームネットワーク環境について整理する．ホームネットワーク環境とは，ユビキタスコンピューティング環境の中でも，特に屋内空間においてネットワークに接続された機器同士が協調動作を行う環境である．本節では，はじめにホームネットワーク環境のハードウェア環境についてまとめ，次にソフトウェア環境について述べる．最後にホームネットワーク環境の課題について考察する．

2.1 ハードウェア環境

ホームネットワーク環境におけるハードウェアの主な構成要素は，ユーザ側にユーザ端末，環境側に情報家電機器，センサの計3つである．

ユーザ端末は固定端末もしくは携帯端末で，ユーザが直接操作する端末である．ユーザは，ユーザ端末から遠隔の機器やセンサに接続する．

家電機器は，制御命令を元に物理空間へ出力を行う機器である．家電機器の中には，AV機器のようにコンテンツを要するものもある．情報家電とは，制御命令や利用するコンテンツをネットワークを介して送信できる家電機器を指す．情報家電機器は，ホームネットワーク内でIP，IEEE1394などのネットワークプロトコルにより相互に接続される．

センサとは，物理空間上の情報を取得し，テキスト情報として出力する機器を指す．ホームネットワーク環境においては，センサはハードウェアが環境側に埋め込まれた形で存在し，ユーザはセンサの出力したテキスト情報をネットワークを介して取得できる．各センサは，超音波や赤外線といったユーザや対象物の位置情報を取得するためのロケーションセンサと，温度や湿度などの環境情報を取得するための環境センサに分類できる．

2.2 ソフトウェア環境

既存の環境において，家電機器上で動作するソフトウェアは，制御命令を物理出力に変換するアプリケーションのみであった．センサについても，物理量の入力をテキスト形式の出力に変換するアプリケーションのみであった．このようなアプリケーションを，本稿ではコアアプリケーションと呼ぶ．これに対し，ホームネットワーク環境ではユーザ端末，情報家電，センサがネットワークを介し相互に接続されることで複数の機器上で動作する分散アプリケーションが実現できる．分散アプリケーション

のアーキテクチャを図1に示し，以下で説明する．

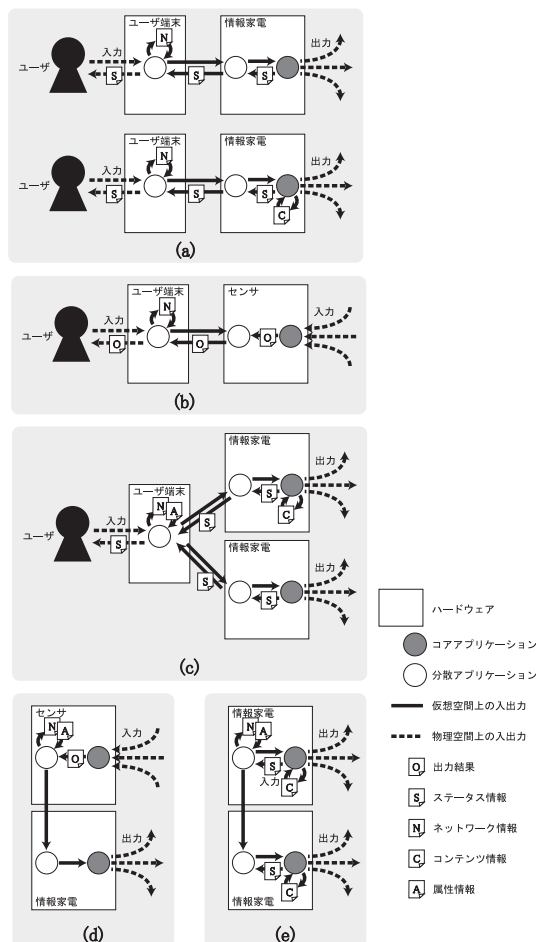


図1 分散アプリケーションのアーキテクチャ (a) 情報家電の単純操作 (b) センサ値取得 (c) ユーザのひとつの入力により複数の機器を操作 (d) センサ値をもとに情報家電を操作 (e) ある情報家電のステータスやコンテンツを元に別の情報家電を操作

図1(a)は，情報家電をユーザ端末から操作するケースである．ユーザ端末と情報家電上では，コアアプリケーションに接続し操作命令を送信するための，分散アプリケーションが動作する．ユーザは，ユーザ端末上の情報家電遠隔操作アプリケーションに操作すべき情報家電のIPアドレスなどのネットワーク情報を入力し，情報家電に接続する．接続後，ユーザ端末から操作命令を情報家電あてに送信すると，情報家電は物理出力を行う．この際，情報家電には物理出力だけでなく，物理出力の状況を仮想空間上で把握するためのステータス情報がユーザ端末あてに返されることがある．たとえば，今ライトがONであるというテキスト情報が返される．情報家電のうち，AV機器やプリンタは単体では動作せず，ビデオテープ，ファイルといったコンテンツ情報が必要となる(図1(a)下図)．こうした情報家電のなかには，ネットワークプリンタのように，コンテンツ情報をユーザ端末から送信することができるものもある．

図1(b)は，ユーザ端末からホームネットワーク内のセンサに接続し，環境情報を取得する場合である．ユーザはユーザ端末上のアプリケーションに，センサのネットワーク情報を入力し，センサと通信を行う．センサからは，物理量である環境情報をテキスト情報に変換した出

表1 アプリケーションプロファイル

| | 情報家電 | センサ | 例 |
|----------|------|-----|-------------------|
| ネットワーク情報 | ○ | ○ | IP アドレス |
| 属性情報 | ○ | ○ | 家電, センサの種類 |
| 出力結果 | × | ○ | センサ値 |
| ステータス情報 | ○ | × | ビデオの再生状態 |
| コンテンツ情報 | ○ | × | ビデオが参照している音楽ファイル名 |

アプリケーションが各アプリケーションプロファイルを必要とする場合があれば ○, なければ ×

力結果がユーザに返される。本アプリケーション例としては、環境センサを使った環境モニタリングがある。

図1(c)は、複数の情報家電によってひとつのアプリケーションが実現される場合である。ユーザは、ユーザ端末からひとつの情報家電を制御するのと同じ感覚で実際には複数の情報家電を制御することができる。制御する情報家電の組み合わせを決定する際に、ネットワーク情報だけでなく情報家電の名前、種類といった属性情報が必要となる。この分散アプリケーション例として、VNA⁵⁾によって構築されるアプリケーションがある。

図1(d)は、センサと情報家電が協調する分散アプリケーションである。センサの出力結果を元に情報家電への制御命令を決定し、情報家電を制御する。このようなアプリケーションはコンテキストウェアアプリケーション⁶⁾と呼ばれる。コンテキストウェアアプリケーションは、センサの取得する物理量を入力とし、それを元にユーザや環境の状態をコンテキストとして抽出する。その後、抽出したコンテキストに適した挙動をする情報家電を制御する。たとえば、ロケーションセンサを使いユーザが睡眠中というコンテキストを抽出し、睡眠中には自動的にライトを消灯させることができる。アプリケーション起動の際には、センサと情報家電のネットワーク情報のほかに、センサの種類、情報家電の種類といった属性情報が必要となる。

図1(e)は、ある情報家電が別の情報家電からの入力を元に制御される分散アプリケーションである。情報家電のコンテンツ情報を同種情報家電間で転送したり、異種情報家電間でコンテンツ変換して転送する⁷⁾アプリケーションがある。また、情報家電のステータス情報を元に、別の情報家電で作業を継続するアプリケーションも提案されている⁸⁾。

以上で示した分散アプリケーションの起動には、いずれの場合もアプリケーションが参照するネットワーク情報、ステータス情報などの情報が必要となる。本稿では、ネットワーク情報、属性情報、出力結果、ステータス情報、コンテンツ情報の5つの情報をまとめてアプリケーションプロファイルと呼ぶことにする。情報家電、センサを利用した分散アプリケーション起動の際に、それぞれ必要となるアプリケーションプロファイルを表1にまとめる。情報家電は、分散アプリケーション起動の際に、アプリケーションプロファイルのうちネットワーク情報、属性情報、ステータス情報、コンテンツ情報が必要となる。一方センサは、ネットワーク情報、属性情報、出力結果の3種が必要となることがある。

2.3 ホームネットワーク環境の課題

前項にて、ホームネットワーク環境において分散アプリケーションを起動するために、アプリケーションプロファイルが必要であることを示した。本稿ではアプリケーションプロファイルに関する二つの問題を挙げ、説明する。

一つ目の問題はアプリケーションプロファイル取得の即時性である。アプリケーションプロファイルのうち、静

的な情報であるネットワーク情報と属性情報は、ディレクトリサービスを通じてテキスト形式で検索することができる。一方、出力結果、ステータス情報、コンテンツ情報は時間の経過に伴い動的に変化する。これらの情報は直接、情報家電やセンサと通信し取得することができるが、動的に変化するため、ユーザが取得したいと思った時点で即座に取得できなければならない。

二つ目の問題は、物理空間の情報家電やセンサと仮想空間上のアプリケーションプロファイルのマッピングである。取得するアプリケーションプロファイルが実際のどの情報家電、センサを指すのかということをユーザが把握できなければならない。DNSや既存のディレクトリサービスでは、たとえばどの情報家電がどのネットワーク情報や属性情報とマップされるのかをあらかじめユーザが把握しておかなければならない。しかし、ホームネットワーク環境においては、数多くの情報家電、センサがあり、それらすべてについてネットワーク情報だけでもあらかじめ把握しておくのは困難である。

以上のアプリケーションプロファイル取得に関する問題により、ホームネットワーク環境では、ユーザが即時に、そして簡単に分散アプリケーションを起動することが困難となると考えられる。ユーザが「あの情報家電を操作したい」と思った場合に、正しいアプリケーションプロファイルをいかに取得し、正しいアプリケーションを起動するかが大きな課題である。この課題を解決し分散アプリケーションの起動に関する即時性と容易性（または直感性）を実現することを、本研究の目的とする。

3. Snap and Use による分散アプリケーションの起動

前節の目的を達成する手段として、対象物を写真で撮影 (Snap) すると同時にアプリケーションプロファイルを即時に取得し、写真閲覧 (Use) によって直感的に分散アプリケーションを起動することを提案する。写真による分散アプリケーション起動イメージを図2に示す。ユーザがユーザ端末上で、写真を閲覧し、写真中のテレビをクリックすると写真が示す実際のテレビを操作するための分散アプリケーションが起動する。撮影した写真が、即座に分散アプリケーションのコンソールとして利用できるのである。

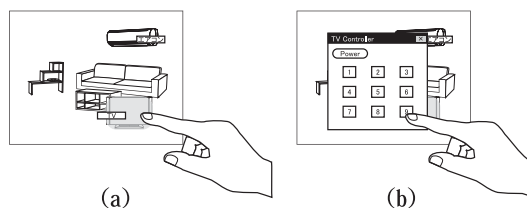


図2 写真による分散アプリケーション起動イメージ (a) ユーザが画像中のテレビをクリックする (b) テレビ操作アプリケーションが起動する

本節ではまず、デジタルメディアへの情報記録について利点をまとめ、次に我々が提案するデジタル写真メディアである u-Photo の機能要件についてまとめる。

3.1 デジタルメディアへの情報記録

現在、デジタルマルチメディアコンテンツにメタデータを新たに埋め込む提案が、数多くなされており、その必要性、有用性が広く認識されている。たとえば、動画内にその動画に関するメタデータを埋め込むことで情報

検索を容易に実現する MPEG-7⁹⁾ や、音楽ファイル内に曲名や歌詞を埋め込むことのできる形式である。

デジタル写真メディアに関しても、PNG 形式、GIF 形式、JPEG 形式等の画像形式にはコメント領域が用意されており、撮影したカメラの情報、撮影場所の GPS 情報等を記録することができる。デジタル写真メディアにメタデータを記録する利点は、記録したメタデータの示す場所やものが写真という視覚情報とともに保存されることで、メタデータ、視覚情報相互のユーザ認知を向上できる点である。

3.2 システム要件

u-Photo は、写真ファイルのコメント領域に、アプリケーションプロファイルを記録した写真ファイルとして設計する。本項では、デジタル写真メディアの特徴と照らし合わせ、デジタル写真メディアにアプリケーションプロファイルする際の要件をまとめる。

画像とアプリケーションプロファイルとのマッピング

u-Photo に付加されるべきアプリケーションプロファイルは、写真画像が示す機器や場所上の情報でなければならない。仮想空間の情報であるアプリケーションプロファイルを物理空間の視覚情報に重ね合わせて示すことで、ユーザはより直感的にアプリケーションプロファイルを認知できる。

写真フレームにあわせた可変数の情報記録

デジタル写真メディアは複数の対象物をひとつのフレームに収めて撮影したり、逆にひとつの対象物だけに限定して撮影することができ、撮影対象の範囲選択をカメラを使って自由に行うことができる。こうした撮影対象の範囲選択に合わせ、可変数のアプリケーションプロファイルを記録することができなければならない。たとえば、複数個数の機器を一枚の写真に撮影した場合は、写っているすべての機器のアプリケーションプロファイルを記録する必要がある。さらに写真中の複数の機器に関するアプリケーションプロファイルは、それぞれが画像中のどの機器を示すかを区別できるように記録する必要がある。撮影時刻のステータス情報、出力結果、コンテンツ情報の記録

デジタル写真メディアは映像メディアコンテンツの中でも、ビデオやリアルタイムストリーミングビデオとは異なり、撮影対象の撮影時刻の視覚状態のみを記録するという特徴がある。動的なものを撮影対象とした場合、対象物のある一瞬の特徴的な状態を静止画として保存できる。この特徴にあわせ、動的に変化するステータス情報、出力結果、コンテンツ情報の撮影時刻に限定した情報を記録する。

情報の即時記録

デジタル写真メディアはカメラを使ってシャッターを押すという簡単な行為のみで作成することができる。u-Photo の生成は、それにあわせ、撮影時に即時にそして自動的に実行されなければならない。ユーザが写真撮影後に写真にアプリケーションプロファイルを入力するのではなく、撮影と同時に自動的にアプリケーションプロファイルを取得し写真に記録する必要がある。

以上の要件を満たすには、メディアである u-Photo のみでは実現せず、u-Photo を閲覧、作成するサブシステムを要する。サブシステムである u-Photo Viewer、u-Photo Creator と u-Photo の関係を図 3 にて示す。ユーザの写真撮影という行為により、u-Photo Creator が u-Photo を生成する。生成した u-Photo は、u-Photo Viewer を

使って開くことにより、アプリケーションプロファイルを付加した形で表示される。u-Photo、u-Photo Viewer、u-Photo Creator により実現される分散アプリケーション起動ツールを u-Photo ツールと呼ぶ。

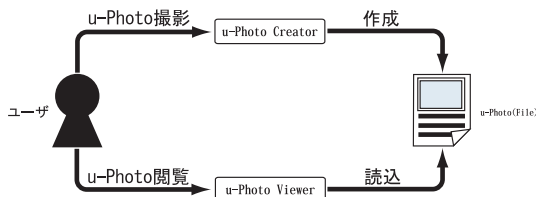


図 3 u-Photo ツール

4. u-Photo ツールのソフトウェアアーキテクチャ

本節では、u-Photo ツールのソフトウェアアーキテクチャについて u-Photo、u-Photo Viewer、u-Photo Creator の順に述べていく。

4.1 u-Photo メディアアーキテクチャ

ファイルとしての u-Photo は、画像ファイルのコメント領域に画像にマップしたアプリケーションプロファイルを示す XML 形式のデータを挿入したファイルとして設計した。u-Photo の DTD を図 4 に示し、以下で説明する。

```
<?xml version="1.0"?>

<!ELEMENT u_photo (location_info, timestamp, focusing_area)>
<!ATTLIST u_photo xsize CDATA #REQUIRED>
<!ATTLIST u_photo ysize CDATA #REQUIRED>

<!ELEMENT location_info (#PCDATA)>
<!ELEMENT timestamp (#PCDATA)>

<!ELEMENT focusing_area (devices, sensors)?>
<!ELEMENT devices (device)+>
<!ELEMENT sensors (sensor)+>

<!ELEMENT device (coordinate, app_profile)*>
<!ATTLIST device id CDATA #REQUIRED>
<!ATTLIST device name CDATA #REQUIRED>
<!ELEMENT coordinate (x, y)>
<!ELEMENT x (#PCDATA)>
<!ELEMENT y (#PCDATA)>

<!ELEMENT sensor (app_profile)*>
```

図 4 u-Photo の DTD

u_photo タグは、timestamp タグと location タグ、focusing_area タグを要素として持つ。timestamp タグは撮影時刻、ロケーションタグはロケーションセンサによって取得されたユーザの撮影場所が記述される。

視覚情報は XML において、写真のフレームを示す一つの focusing_area タグと、写真に写っている情報家電を示す device タグとして表現される。写真内に複数の情報家電がある場合、device タグは複数記述され、devices タグによってまとめられる。写真内の情報家電はすべてフレームに収められているため、devices タグは focusing_area の要素として扱われる。

focusing_area は、device タグの他にフレームに写っている場所に埋められているセンサに関する sensors タグを要素として持つ。sensors タグの要素である sensor タグには、アプリケーションプロファイル記述用のタグが

```

<wapplet name="CDPlayer">
  <media_type>audio</media_type>
  <status>10000</status>
  <time>0</time>
  <service_provider>AudioProvider</service_provider>
  <ip>192.168.10.4</ip>
</wapplet>

```

図 5 app_profile タグの記述例

記録される。図中では、app_profile としているが、センサ上で動作するアプリケーションを示す名前のタグとして使われる。ひとつのセンサ上で複数のアプリケーション起動ができる場合、アプリケーションの数だけこのタグは記録される。このタグには、アプリケーション起動に必要なアプリケーションプロファイルが記録される。

device タグにも、同様に情報家電上で動作する分散アプリケーション用のアプリケーションプロファイルが記録される。app_profile タグの例を図 5 に示す。このタグは、wapplet という app_profile タグで、wapplet にとってのアプリケーションプロファイルを記述したタグを要素として持つ。このように各アプリケーションは、app_profile タグで独自のアプリケーションプロファイルを記述する。

さらに、device タグでは、画像中の情報家電の位置を示すため、画像中の x 座標、y 座標を表す coordinate タグが加えられる。

4.2 u-Photo Viewer

u-Photo Viewer は u-Photo を閲覧する GUI をユーザに提供する。u-Photo Viewer の提供する GUI の機能は、表示した写真画像中の情報家電をマウスやタッチパネルでクリックすることで、アプリケーション用のメニュー GUI を提示することである。通常の分散アプリケーションでは、アプリケーションを起動した後に接続先のネットワーク情報などのアプリケーションプロファイルを入力しなければならない。一方 u-Photo Viewer では、u-Photo 中の情報家電のクリックという動作によって、アプリケーションプロファイルが自動的に入力された状態でアプリケーションが起動する。

u-Photo Viewer のシステムアーキテクチャを図 6 にて示す。u-Photo Viewer によって u-Photo ファイルを開くと、XML 解析機構が u-Photo に記録された XML を解析し、u-Photo 基本 GUI を表示する。u-Photo 基本 GUI には、各 device タグ内の coordinate タグの示す座標に基づいて、画像中の各情報家電上にアイコンが表示されている。画像上のアイコンをクリックすると、その情報家電上で動作するアプリケーションのリストが GUI 上に表示される。これは、devices タグ内の app_profile タグによって決められる。ユーザが起動させるアプリケーションを選択すると、イベント解析機構が app_profile タグで指定されたアプリケーションに app_profile タグで記述されたアプリケーションプロファイルを渡す。

各アプリケーションは、情報家電の IP アドレスなどのアプリケーションプロファイルが入力された状態でアプリケーションの GUI を表示する。ユーザは各アプリケーションの GUI を使って情報家電上の分散アプリケーションを利用できる。以上の動作は第 3 節の図 2 の動作を実現するものである。

イベント解析機構では、GUI 上の情報家電機器を示すアイコンのクリックだけでなく、アイコンへのファイル

のドラッグアンドドロップ、アイコン同士のドラッグアンドドロップといったイベントも取得し、分散アプリケーションを起動する機能も持つ。ユーザ端末上のファイルを情報家電のアイコン上に置くと、情報家電がプリンタや AV 機器といったコンテンツ情報を要するものでかつファイルが印刷用ファイル、ビデオファイルなどであるならば、操作したファイルをコンテンツ情報としてアプリケーションを起動させることができる。また、2 つの情報家電のアイコン同士をドラッグアンドドロップすると、情報家電の機能組み合わせアプリケーション、複数の情報家電をまたがった作業継続アプリケーションなど、複数の情報家電を協調させるアプリケーションのメニューを呼び出すことができる。

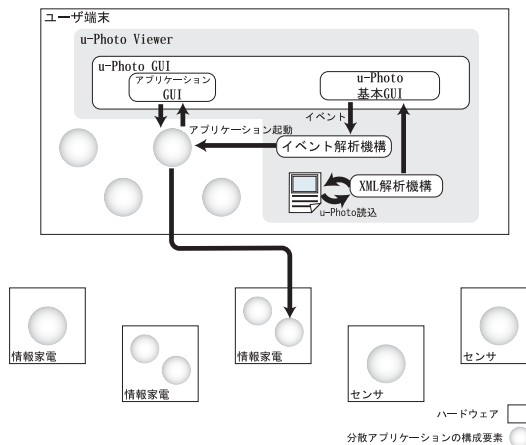


図 6 u-Photo Viewer のシステムアーキテクチャ

4.3 u-Photo Creator

u-Photo Creator は、写真撮影時に即時に撮影した画像内のアプリケーションプロファイルを取得し、u-Photo ファイルとして保存するシステムである。本項では、u-Photo Creator システム実現時に課題となる写真内の情報家電およびセンサの識別方法について説明し、その後システムアーキテクチャについて述べる。

4.3.1 画像中の情報家電、センサの識別方法

u-Photo において、画像によって表現される視覚情報は focusing_area と device という二つに区別されることを先に述べた。画像全体のフレームを表す focusing_area には、実際にその場所に存在するセンサのアプリケーションプロファイルが埋め込まれる。したがって、u-Photo において画像中で区別すべき対象物は情報家電とセンサである。画像中の情報家電とセンサを何らかの形で識別しなければならない。

u-Photo Creator では、画像中の情報家電認識方法として、画像解析を行う。画像解析には、解析する対象物に 2 次元バーコード、LED タグといった固有の ID を固有の視覚情報として表示することのできるタグを取り付け画像中のタグを元に解析する方法と、対象物そのものの形を登録しておき、その情報を元にタグを取り付けることなく解析する方法がある。

タグを取り付けない場合、画像解析の元となる画像を複数個あらかじめ登録しておかなければならない¹⁰⁾ うえ、同じ形の情報家電が複数ある場合にはそれらを区別して認識するのが難しい。したがって u-Photo Creator では、2 次元バーコード、LED タグといった画像認識用タグに固有の ID を割り当て、画像解析により ID と、ID

の存在する座標を取得する方法を採用する。タグの ID は、情報家電のネットワーク情報、属性情報と照合された形でディレクトリサービスに保存され、ID を元にネットワーク情報、属性情報が取得できることを想定する。

センサの場合はより複雑である。センサは環境側に埋め込まれ見えない形で存在することを想定するため、画像解析によりセンサの存在を取得するのは難しい。また、写真は 3 次元の空間情報を 2 次元の画像に変換して捉えるため、センサの埋め込まれている場所を取得し画像上に表示しても、ユーザが正しくセンサの所在地を把握するのは困難である。

u-Photo では、環境側に埋め込まれたセンサの情報取得方法として、実空間上の位置情報を利用した二つの方法を提案する。一つ目は、撮影場所を取得し、撮影場所にあるセンサの出力情報を、写っているいないにかかわらずフレーム内の出力情報として扱う方法である。たとえば、撮影する部屋の場所をロケーションセンサによって取得し、部屋にある環境センサの出力情報を、室内のセンサの出力情報として保存する。二つ目は写真内の情報家電の情報をもとに、情報家電の周囲にあるセンサの出力情報をフレーム内の出力情報として扱う方法である。u-Photo Creator では、情報家電が写真内にある場合には後者の方法、ない場合には前者の方法で取得するセンサを指定するが、いずれの場合も写真フレームが示す場所にあるセンサを正確に指定することは難しい。

4.3.2 システムアーキテクチャ

u-Photo Creator のシステムアーキテクチャを図 7 にて示し、以下で図中の各機構について説明する。

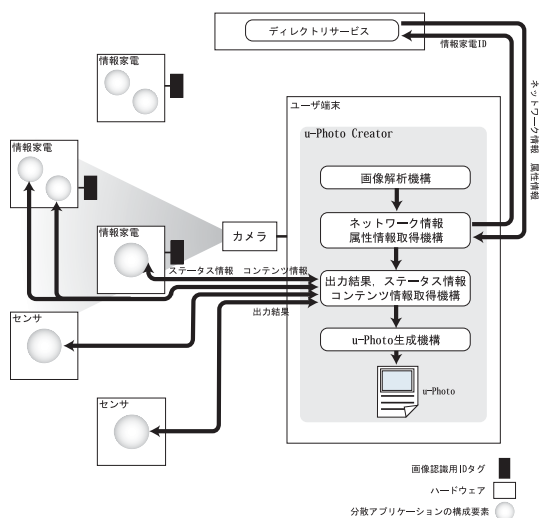


図 7 u-Photo Creator のシステムアーキテクチャ

機器 ID 取得機構

カメラから画像を取得し、画像内にある情報家電の ID と画像中の座標を画像解析によって取得する機構である。取得された情報家電の ID と画像中の座標はネットワーク情報、属性情報取得機構へ渡される。ネットワーク情報、属性情報取得機構

機器 ID 取得機構から渡された情報家電の ID をもとに、ディレクトリサービスへアプリケーションプロファイルのうち静的な情報であるネットワーク情

同種のセンサが複数存在する場合は、複数の出力情報の平均値を採用するなどの手段をとる

```
<UI name="Light">
<status>OFF</status>
<button name="ON">
<ip>192.168.10.6</ip>
<port>12345</port>
<command>LIGHT_ON</command>
</button>
<button name="Get Status">
<ip>192.168.10.6</ip>
<port>12345</port>
<get_command>GET_STATUS</get_command>
</button>
</UI>
```

図 8 ライト用 app_profile タグ

報、属性情報を機器を問い合わせる。センサのネットワーク情報、属性情報もこのとき同時にディレクトリサービスに問い合わせる。取得した情報家電およびセンサのネットワーク情報、属性情報は出力結果、ステータス情報、コンテンツ情報取得機構に渡される。

出力結果、ステータス情報、コンテンツ情報取得機構渡されたネットワーク情報、属性情報が示す情報家電、センサあてに、そこで動作するアプリケーションの動的情報である出力結果、ステータス情報およびコンテンツ情報を問い合わせる。情報家電からは、現在のステータス情報およびコンテンツ情報を取得し、センサからは現在の出力結果を取得する。

u-Photo 生成機構

上記の機構それぞれで取得された情報をすべてまとめ、XML 形式で保存し画像ファイルにまとめる機構である。本機構を経て u-Photo が生成される。

5. プロトタイプ実装

本節では u-Photo ツールのプロトタイプ実装について述べる。

5.1 実装環境

プロトタイプ実装では、カメラに Logitech 社製 Web カメラを、各情報家電の画像認識用 ID タグとして色と点滅パターンによって ID を表現する LED タグを使用した。u-Photo Creator および u-Photo Viewer は、PC 用、Zaurus SL-C760 用の 2 バージョンを Java を用いて実装した。センサ情報を取得するセンサとして、小型センサユニット mica2¹⁵⁾ を 8 個使用した。

5.2 アプリケーション

u-Photo を使ったアプリケーションとして、ライトとセンサの制御アプリケーション、Wapplet を使ったビデオとプリンタのアプリケーションを実装した。

5.2.1 単純制御アプリケーション

制御命令を対象機器あてに送信する制御アプリケーションを実装した。このアプリケーションを使い、ライトの機器制御を行った。u-Photo XML 中のライト用 app_profile タグを図 8 で示し、ここではライトの制御について説明する。図 8 status タグにて撮影時のライトのステータス情報が示される。button タグは、アプリケーション GUI 上に、name で示す名前のボタンを配置する。ユーザがボタンをクリックすると、ip タグ、port タグで示す IP アドレス、ポート番号宛に command タグに記入されたコマンドを送信し、ライトを制御する。

センサ情報を取得するためのシステムとして MARS¹¹⁾ サーバを利用した。MARS サーバは、複数のセンサを位

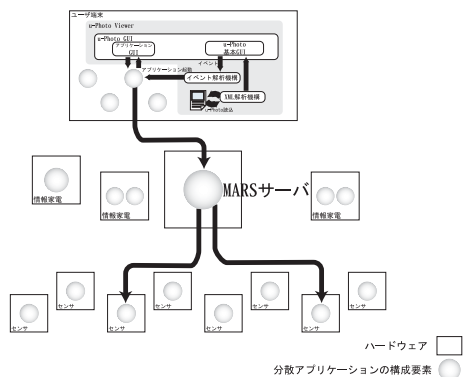


図9 MARSサーバ

置情報とともに管理する．MARSサーバを利用するアプリケーションは一箇所のMARSサーバにクエリを送信するだけで、複数のセンサから必要な情報を取得することができる．MARSサーバを利用するアプリケーションのapp_profileタグは、図8とほぼ同様の構造を持ち、撮影時のセンサの出力結果と、現在のセンサの出力結果を取得するためのボタンがGUI上に表示される．

図10は、PDA上のu-Photo Viewerでライトを撮影したu-PhotoのGUI(a)と実際のライト制御の様子(b)である．

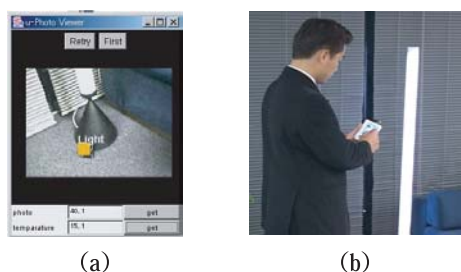


図10 ライト操作アプリケーション (a)PDA用u-Photo Viewer (b)ライト操作の様子

5.2.2 Wapplet アプリケーション

機器の協調動作のためのアプリケーションフレームワークであるWapplet⁸⁾を利用したアプリケーションとして、プリンタ、ビデオアプリケーションを実装した．

プリンタアプリケーションでは、PC上のu-Photo Viewerのイベント解析機構により、ファイルから写真中のプリンタへのドラッグアンドドロップによって、印刷するファイル指定を実現した(図11(a))．

ビデオアプリケーションでは、ビデオ再生中のディスプレイをu-Photoで撮影し、再生状態をWappletのアプリケーションプロファイルとして保存する(図11(b))．その後、u-Photo ViewerのWappletメニューから、ビデオの続きを見る別機器を指定しその機器上で同じビデオを撮影時の続きから見る事ができる(図11(c))．

6. 基本性能評価

u-Photo ツールの基本性能評価として、u-Photo 生成にかかる時間をu-Photo Creator内の各機構ごとに測定した．u-Photo Creatorの測定に使用したハードウェアおよびソフトウェアの環境を表2に示す．

ディレクトリサービスはu-Photo Creatorと同じ端末上で立ち上げ、ネットワークを介さない時間として測定

表2 測定環境

| | |
|--------|------------------|
| CPU | PentiumM 1.70GHz |
| 主記憶 | 1GB |
| OS | WindowsXP |
| JavaVM | J2SDK1.4.2 |

する．同様に、u-Photo Creatorと同じ端末上でビデオを再生するアプリケーションを立ち上げ、これを撮影対象とした．た．ビデオのステータス情報、コンテンツ情報はWappletによって管理される．

1枚のu-Photo撮影につき、u-Photo Creator内の各機構で要した時間とその合計時間について、40回測定し、その平均時間を表3に示す．

表3 u-Photo生成時間測定結果(ms)

| 処理内容 | 処理時間(ms) |
|--------------------------|----------|
| 画像解析機構 | 71.1 |
| ネットワーク情報、属性情報取得機構 | 9.8 |
| 出力情報、ステータス情報、コンテンツ情報取得機構 | 12.3 |
| u-Photo生成機構 | 23.8 |
| Total | 117 |

測定の結果、u-Photo生成にかかる時間は全体として平均117msかかることがわかった．この数値は、写真撮影という一連の動作の中でユーザが遅延を感じる事のない、十分な短時間であるといえる．ただし、本測定では、ディレクトリサービスおよびアプリケーションをu-Photo Creatorと同端末上で立ち上げているため、実際には、それぞれとのネットワークの通信にかかるRTTが加算される．しかし、ディレクトリサービスおよびアプリケーションはすべて同一ホームネットワーク上にあるため、通信時間は短時間であり、ユーザが遅延を感じるには至らないと考えられる．また、画像解析機構での処理時間がもっとも長く平均71.1msであったが、今後画像解析手法について検討していく中で、より短い時間の画像解析が実現する可能性がある．

7. 関連研究との比較

関連研究として、NaviCam¹²⁾、InfoPoint¹³⁾、DigiScope¹⁴⁾が挙げられる．

NaviCamは、ユーザが携帯するCCDカメラから取得した映像に、写っているものに関するメッセージを付加し、ヘッドマウントディスプレイやPDA上にリアルタイムで表示するシステムである．対象物の画像上の認識には、2次元バーコードを利用している．見えない情報を画像情報上に視覚化するという点において、NaviCamの目的はu-Photoと共通している．しかしNaviCamでは、対象物上で動作するアプリケーションの起動は実現していない．

InfoPointは、スティック型のデバイスで、情報家電上につけられた2次元バーコードに向けることで情報家電上のアプリケーションを起動させることができる．また、スティック型デバイスを使い、複数の情報家電間でドラッグアンドドロップ、ポイントアンドクリックといった動作により、複数の情報家電間の分散アプリケーションを起動することができる．InfoPointの情報取得は、スティック型デバイスを情報家電に向けるという、ユーザと情報家電の1対1のアプローチである．一方u-Photoでは、カメラにより、複数の情報家電を選択することもでき、それを写真として後に閲覧、操作することで、ユーザと情報家電さらに範囲内のセンサの1対多の情報取得が可能である．

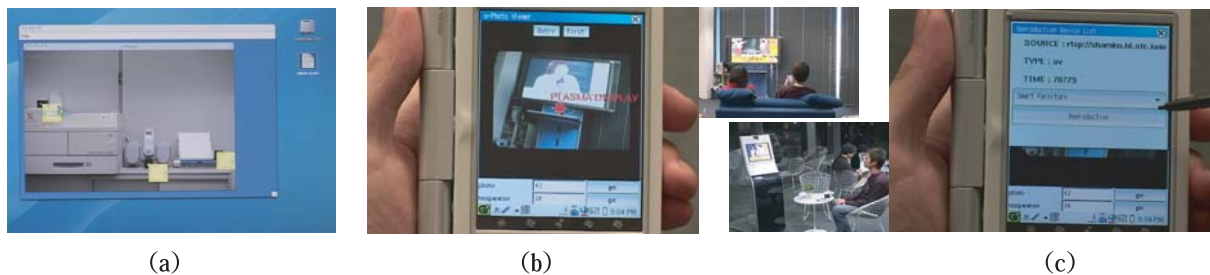


図 11 Wapplet によるアプリケーション動作 (a) プリンタアイコンへのファイルのドラッグアンドドロップによる印刷 (b)u-Photo 撮影によるビデオ再生状態の取得 (c) 別機器へのビデオ再生状態の作業継続

DigiScope は、シースルーのタブレット型デバイスを使い、タブレットを通じて見える景色上の情報家電などのオブジェクトを操作する GUI をタブレット上に表示するシステムである。DigiScope を通じて見るオブジェクト上の分散アプリケーションを起動でき、かつ一対多の情報取得が実現できる。DigiScope と u-Photo の相違点は、取得した情報の保存による 2 次的利用である。u-Photo は、情報家電のステータス情報などの動的な情報を保存しておき、後に参照したり、一度撮影した u-Photo を使った後から情報家電を遠隔操作したりすることができる。

8. おわりに

本稿では、写真撮影によってホームネットワーク環境におけるアプリケーションの情報取得し、撮影したデジタル写真を通じて直感的に情報閲覧やアプリケーション起動を行う u-Photo ツールを提案し、設計、実装および基本性能の評価を行った。

u-Photo の今後の課題として、過去の u-Photo 利用に際する環境との一貫性の確認と、ユーザごとのパーソナライズがある。u-Photo はアプリケーションプロファイルメディアとして保存するため、後からの閲覧や利用が可能である反面、情報家電やセンサを移動させたり、新たに追加することによってハードウェアの設置環境が変化した場合、現在の u-Photo ツールでは対応できない。環境側に、過去の u-Photo と現在の環境との一貫性を確認するための機構をあらたに構築する必要がある。また、u-Photo ツールは現在撮影や閲覧するユーザを区別していない。誰が撮影しても誰が閲覧しても同じ u-Photo である。しかし、家庭内の情報家電やセンサはユーザによってパーソナライズされ、利用権限などが決められることも考えられる。u-Photo のユーザごとにパーソナライズしていく機構の開発も今後の課題である。

参 考 文 献

- 1) Sun Microsystems Inc.: Jini Technology Overview (1998).
- 2) The Havi Organization: Home Audio Video Interoperability (1998). <http://www.havi.org>.
- 3) Universal Plug and Play Forum. Universal Plug and Play (UPnP), 1999. <http://www.upnp.org>.
- 4) T. Okoshi, S. Wakayama, Y. Sugita, S. Aoki, T. Iwamoto, J. Nakazawa, T. Nagata, D. Furusaka, M. Iwai, A. Kusumoto, N. Harashima, J. Yura, N. Nishio, Y. Tobe, Y. Ikeda, and H. Tokuda, "Smart Space Laboratory Project: Toward the Next Generation Computing Environment.," In *Proceedings of IEEE Third Workshop on Networked Appliances*

- (IWNA), Feb. 2001
- 5) J. Nakazawa, T. Okoshi, M. Mochizuki, Y. Tobe, and H. Tokuda. "VNA: An Object Model for Virtual Network Appliances.," In *Proceedings of IEEE International Conference on Consumer Electronics (ICCE2000)*, June 2000.
- 6) B. Schilit, N. Adams, and R. Want, "Context-Aware Computing Applications.," In *Proceedings of the 1st International Workshop on Mobile Computing Systems and Applications*, pp. 85-90. 1994.
- 7) R. Mohan, J. Smith, C.-S. Li, "Adapting Multimedia Internet Content For Universal Access" *IEEE Transactions on Multimedia*, pp. 104-114. 1999.
- 8) T. Iwamoto, N. Nishio, and H. Tokuda. "Wapplet: A Media Access Framework for Wearable Applications.," In *Proceedings of International Conference on Information Networking*, volume II, pp. 5D4.1-5D4.11. 2002.
- 9) F. Nack, A. Lindsay, "Everything you wanted to know about MPEG-7", *IEEE Multimedia*, Vol. 6, No. 3, pp. 65-77 (Part II in Vol. 6, No. 4). 1999.
- 10) I. Haritaoglu. "Infoscope: Link from Real World to Digital Information Space.," In *Proceedings of the 3rd International Conference on Ubiquitous Computing*, pp. 247-255. 2001.
- 11) 丸山大佑, 青木俊, 高汐一紀, 徳田英幸, "センサ飲めた情報を利用したセンサデータ取得ミドルウェア.," 第 4 回 コピキタスコンピューティングシステム研究会, 2004 年 4 月.
- 12) J. Rekimoto and K. Nagao, "The World Through the Computer: Computer Augmented Interaction with Real World.," In *Proceedings of Symposium on User Interface Software and Technology*, pp. 29-36. 1995.
- 13) N. Kohtake, J. Rekimoto, and Y. Anzai, "Info-Point: A Device that Provides a Uniform User Interface to Allow Appliances to Work Together over a Network.," *Personal and Ubiquitous Computing*, vol5, no.4, pp.264-274, 2001.
- 14) A. Ferscha and M. Keller, "Digiscope: An Invisible World Window.," In *Adjunct Proceedings of The Fifth International Conference on Ubiquitous Computing*, pp.261-262. acm, 2003
- 15) J. Hill and D. Culler, "A Wireless Embedded Sensor Architecture for System-Level Optimization.," In *UC Berkeley Technical Report*.