

# ユビキタス環境構築のためのブロック型情報機器連携技法

神 武 直 彦<sup>†1,†2</sup> 大 澤 亮<sup>†3</sup> 米 澤 拓 郎<sup>†3</sup>  
高 汐 一 紀<sup>†4</sup> 徳 田 英 幸<sup>†3,†4</sup>

ユビキタスコンピューティング環境の普及にともない、その環境を利用者が自分自身で構築できることが重要になってきている。その実現のためには、利用者の認知的負荷を増加させることなく複数の情報機器を連携させ、ユビキタスコンピューティング環境を構築、拡張することが可能なユーザインタフェースが必要である。本論文では、組み立てることで自己組織化するブロック型情報機器 u-Texture II を提案する。利用者は u-Texture II どうしを接続することで様々な形に組み立てることができる。それに対し、u-Texture II は組み立てられた形状を自律的に認識し、形状に対応して機能を変化させ、複数で連携して動作することが可能である。

## Collaboration Scheme Based on Blockable Appliances for Ubiquitous Environment

NAOHIKO KOHTAKE,<sup>†1,†2</sup> RYO OHSAWA,<sup>†3</sup> TAKURO YONEZAWA,<sup>†3</sup>  
KAZUNORI TAKASHIO<sup>†4</sup> and HIDEYUKI TOKUDA<sup>†3,†4</sup>

With diffusion of ubiquitous computing environments, it is essential that non-expert users can create ubiquitous computing environment by themselves. Compared with computer users who are mainly engineers or experts, ubiquitous computing users are varied and a friendly user interface is desired. This paper proposes collaboration scheme based on blockable appliances for ubiquitous computing environment. This collaboration scheme has been implemented in the u-Texture II, which is a self-organizable panel that works as a building block. The u-Texture II can change its own behavior through recognition of its location, its inclination, and surrounding environment by assembling these factors physically. This paper describes the design, the implementation, and various applications of u-Texture II to confirm that the collaboration scheme can contribute to establishment of ubiquitous computing environments without expert users.

### 1. はじめに

携帯端末から家電機器、家具、部屋、建物に至るまで様々なものが計算機能やセンサ機能、通信機能を有した情報機器として機能し、それらが協調することによって利用者の活動を支援す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<sup>1)~3)</sup>。このような環境では、情報機器どうしがデータをやりとりし、連携することにより、利用者は従来のコンピューティング環

境とは異なる形で作業支援を受けることが可能になる。また、ユビキタスコンピューティング環境における情報機器の利用者は子供から老人まで様々であり、多くの利用者が容易に利用可能なユーザインタフェースが必要とされている。

ユビキタスコンピューティング環境の普及にともない、自分で部屋のレイアウトを変更したり、家具の構成を拡張したりするように、利用者が自分自身で情報機器を組み替えることで、ユビキタスコンピューティング環境を自由に構築したり、拡張したりできることが重要になってきている。それにより、利用者は自分の要望に合致した適切な支援を受けることが可能になる。また、その場合、計算機の内容を意識することなく情報機器を扱えることがユビキタスコンピューティング環境の理想である<sup>4)</sup>。

現在、利用者と複数の情報機器とのインタラクションのためのユーザインタフェースは、利用者が1台

†1 慶應義塾大学 SFC 研究所

Keio Research Institute at SFC, Keio University

†2 宇宙航空研究開発機構

Japan Aerospace Exploration Agency

†3 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

†4 慶應義塾大学環境情報学部

Faculty of Environmental Information, Keio University

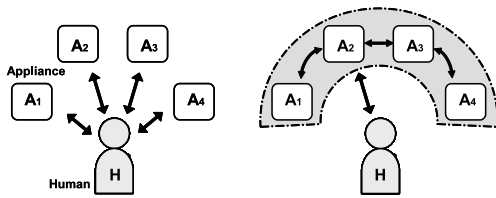


図 1 1 台のコンピュータを利用するための従来のユーザインタフェースと MUI のインタラクション形態の比較

Fig. 1 Comparison of interaction styles between single-computer UI and Multi-appliance UI.

のコンピュータを利用するためのユーザインタフェースがいまだに多く利用されている．そのため，複数の情報機器を連携させ，作業支援を受ける場合には，利用者は機器の接続方法を意識的に記憶したり，連携方法を認識したりしなければならないといった認知的負荷が増大することがある．たとえば，利用者が情報機器の電気的な接続構成やネットワーク上のアドレスやマシン名を記憶したり，必要なデータや入出力インタフェースの所在を認識したりすることで複数の情報機器の連携を可能にしている．

この問題を解決するために，利用者の認知的負荷を増加させることなく複数の情報機器を連携させることが可能なユーザインタフェースが必要である．このインタフェースを Multi-appliance User Interface (以下，MUI) と呼ぶ．図 1 に複数の情報機器連携操作における 1 台のコンピュータを利用するための従来のユーザインタフェースと MUI のインタラクション形態に関する比較を示す．MUI を実現するために考慮すべきことの 1 つは，利用者と情報機器間のインタラクションおよび連携する複数の情報機器間のインタラクションの方法とその負荷である．利用者からの意図的な入力なしに情報機器が利用者の意図を正確に認識して連携できることが理想であるが，すべての状況でそれを実現することは現状の計算機やセンサの技術では不可能である．そのため，利用者が意識的に行わなければならない情報機器への入力を極力少なくし，情報機器がその少ない意識的な入力と利用者による無意識的な動作から利用者の意図を的確に認識できる技術が必要である．また，情報機器どうしが自律的にデータをやりとりして適切に連携できる技術が必要である．

本論文では，ユビキタスコンピューティング環境を構築することを目的とした MUI を実現するシステムの 1 つとして，組み立てることによって自己組織化するブロック型情報機器 u-Texture II (以下，uTXII) を提案する．筆者らは，MUI を実現することを目的として情報機器の連携のための携帯デバイス<sup>5)</sup> やスナツ



図 2 uTXII の組み立て形状の例

Fig. 2 Examples of assembled uTXIIs.

ブショットインタフェース<sup>6)</sup>の研究を実施してきた．これに対し，今後は利用者がユビキタスコンピューティング環境を新たに構築し，拡張するための MUI を実現するシステムが必要である．そのため，本論文では，様々な形に組み立てることができ，容易に形状を変更したり，拡張したりできることが可能なことに注目し，ブロック型情報機器を提案する．

利用者は複数の uTXII を水平，垂直方向に組み立てることができ，それらをテーブルや棚などの家具，または，壁や床などの建物の一部として形作ることができる (図 2)．組み立てられた uTXII は，互いのセンサ情報をやりとりすることで自律的に組み立て形状と自己の位置を認識する．そして，形状に対応して適切なアプリケーションの候補を利用者に提示する．また，あらかじめ実行するアプリケーションが選択されている場合には，そのアプリケーションを実行させるために必要な形状への組み立て手順を動的にナビゲートする．組み立て形状が完成し，アプリケーションが選択された段階で，uTXII どうしが連携してアプリケーションを実行する．利用者にとっては，計算機やセンサの存在を意識することなく uTXII を組み立てるといった動作によって容易にユビキタスコンピューティング環境を構築することが可能になる．

本研究において，現在までにプロトタイプモデルとして u-Texture I (以下，uTXI) の研究を実施してきた<sup>7)</sup>．uTXII では，uTXI の研究成果や予備実験を基に実利用を考慮して設計，実装を行い，利用者の認知的負荷をさらに軽減する MUI を実現した．また，周辺情報機器との連携も考慮した設計とし，uTXII をある家庭や室内公共空間，協働空間に設置して運用を行った．

以下では，uTXII を実現するための具体的な説明を述べ，このシステムを用いて実現したアプリケーションのうち，代表的なものを紹介する．また，ユーザ経

験によって有効性を確認した結果を報告する。

## 2. u-Texture II

### 2.1 設計方針

同じ部材でできた棚や机、壁があった場合、棚は物の設置や保管に利用され、机は利用者単独の作業や他の利用者との会議や協調作業に利用される。また、壁は空間の遮蔽や情報の掲示に利用されるなど、それぞれの形状によってある程度機能が異なる。そのため、部材自体が自己の組み立て形状を認識することができれば、利用者の意図的な入力なしに、その形状に対する利用の意図を予測できると考えた。なお、uTXII はパネル型のブロックであり、組み立てることで家具や建物の一部を形作ることができるが、その形状をレンガ型やタイル型にした場合、組み立て形状やその大きさ、粒度が異なるため、それにとまって利用のされ方が変化する。そのような場合でも組み立て形状に対応した自己認識や連携に関する uTXII の機能はそのまま適用することが可能になるよう汎用性を考慮した設計とした。プロトタイプモデルである uTXI においては、uTXI の質量を支えるための接続機構の組み立てに時間を要してしまうという問題と、利用者が uTXI を実際に組み立てなければ実行可能なアプリケーション候補を確認できないという問題があったが、uTXII ではそれらを解決した設計とした。uTXII の主な設計方針を以下のように設定した。

#### 2.1.1 容易な組み立て

利用者は、内部の計算機やセンサ、ネットワークの存在を認識することなく、容易に uTXII を組み立てられる。また、組み立てには特別な道具は必要なく、原則として uTXII どうしをつなぎ合わせることで組み立てることができる。

#### 2.1.2 組み立て形状の自己認識

組み立てられた uTXII どうしが接続や傾きに関するお互いのセンタデータをやりとりすることで、利用者の意図的な入力なしに組み立て形状やその形状における自己の位置や傾きを認識する。利用者がアクセスポイントやデータサーバなどの uTXII 以外の機器を設置することなしに、uTXII はお互いの直接のやりとりのみで形状の自己認識を行う。

#### 2.1.3 組み立て形状に対応した連携動作

キーボードによるテキスト入力やマウスによるカーソル操作によって連携させたい情報機器のマシン名やアドレスを入力することなしに、利用者が連携させたい uTXII どうしを接続し、組み立てることで、組み立てられた uTXII は連携して適切なアプリケーション

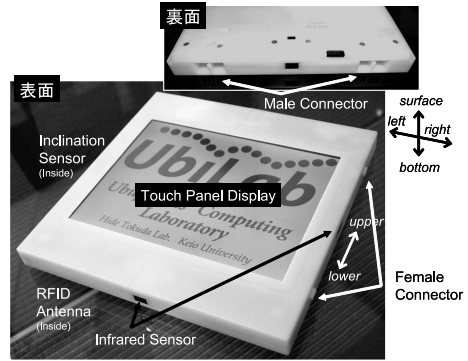


図 3 uTXII の外観

Fig.3 Appearance of a uTXII.

を実行する。また、すべての uTXII の仕様を同一にし、組み立て形状内において uTXII を入れ替えた場合にはそれらの uTXII の機能が自律的に入れ替わる。

### 2.2 機能

uTXII とのインタラクションにおいて、利用者の動作は、uTXII を組み立てることと、組み立てられた uTXII で実行するアプリケーションをどれか 1 つの uTXII で選択することの 2 つである。図 3 に uTXII の外観を示す。縦横 290 mm の正方形の平面であり、厚さは 32.2 mm で、質量は 2,500 g である。内部のデバイスの統合や配線の簡略化、すべての通信の無線化などにより uTXI と比較して約 40%軽量化した。各側面に機械的なコネクタが存在し、特別な道具なしでの uTXII どうしの接続を可能にしている。他の uTXII との接続の有無を検知するための赤外線センサと自己の傾きを検知する 3 次元センサで組み立て形状を認識し、利用者への出力はスピーカとタッチパネルディスプレイで行う。アプリケーション候補の選択などの利用者からの入力もタッチパネルディスプレイで受け付ける。また、RFID アンテナをディスプレイの外枠裏側に設置し、uTXII 上に置かれたものを認識する。

利用者とのやりとりのために、uTXII では Shape Assemble モード（以下、SA モード）と Application Select モード（以下、AS モード）の 2 つのインタラクションモードが存在する。図 4、図 5 にそれぞれのインタラクションモードでの組み立て開始からアプリケーション実行までの流れの例を示す。各図において、左側が利用者の動作の様子であり、右側がそれぞれの動作に対応した uTXII の画面出力の様子である。まず SA モードは、組み立てられた形状に対応し、その形状で実行可能なアプリケーションの候補を提示するモードである。組み立てる前には uTXII 単体で実行可能なアプリケーションの候補が提示され（図 4-1）、

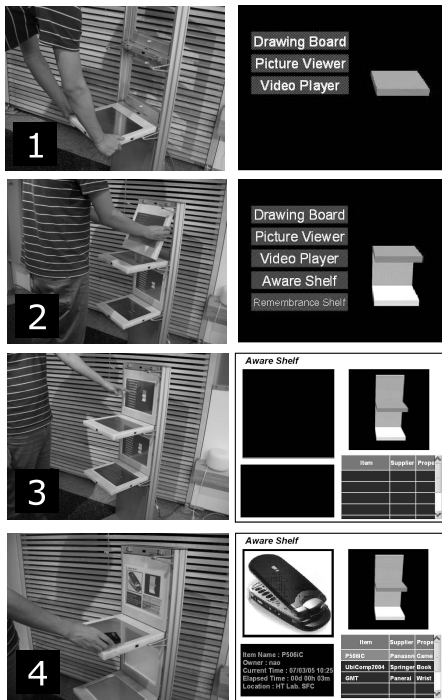


図4 SAモードによるuTXII組み立ての流れ  
Fig. 4 Sequence in SA Mode assembly of uTXIIs.

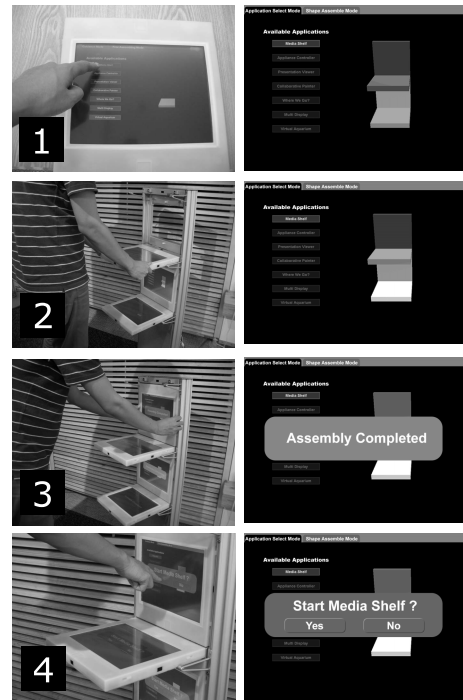


図5 ASモードによるuTXII組み立ての流れ  
Fig. 5 Sequence in AS Mode assembly of uTXIIs.

組み立てていく過程において、uTXIIが認識している組み立て形状とそのuTXII自体の位置が色分けして表示され、組み立て形状に対応したアプリケーションの候補が提示される(図4-2)。そして、利用者が候補の1つを選択することで組み立てられたuTXIIが連携してアプリケーションを起動し(図4-3)、利用者はそのアプリケーションを利用することができるようになる(図4-4)。一方、ASモードは、実行したいアプリケーションを利用者があらかじめ選択することでuTXIIの組み立て方をナビゲートするモードであり、uTXIでの予備実験の経験を基に新たに開発した。組み立てる前、もしくは、組み立ての過程において実行したいアプリケーションを利用者が選択すると組み立てるべき全体形状が提示され(図5-1)、組み立てていく過程において、組み立てが完了している部分とこれから組み立てるべき部分、そのuTXII自体の位置が色分けして表示される(図5-2)。そして、指定された形状にuTXIIが組み立てられるとアプリケーションが実行可能であることが提示され(図5-3)、それを実行するか否かのボタンが提示される(図5-4)。Yesボタンを押すことで実行することができるが、Noボタンを押して組み立てを継続することも可能である。提示される組み立て形状は選択されたアプリケーションを最も少ない台数のuTXIIで実現可能な形状であ

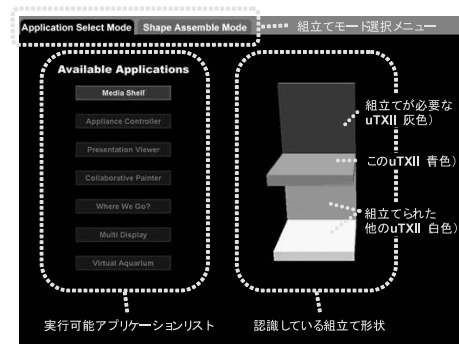


図6 ASモードにおけるuTXIIのディスプレイ表示  
Fig. 6 GUI display on uTXII in AS Mode.

り、組み立てられた形状を認識して提示する組み立て形状のスケールが変化する。図6に画像出力の例として、ASモードにおけるuTXIIのディスプレイ表示を示す。実行可能なアプリケーション候補のリストが左側に表示され、認識されたuTXIIの組み立て形状が右側に表示される。認識された組み立て形状については、ディスプレイ表示をしているuTXII自体が青色で表示され、組み立てられている他のuTXIIは白色で表示される。また、ASモードでは、選択されたアプリケーションを実行するのに追加で組み立てる必要のあるuTXIIが灰色で表示される。左上のモード

メニューを選択することで、インタラクションモードを切り替えることができる。

なお、組み立てた uTXII を垂直に立てる際に重さを支えるのに必要な接続機構については、uTXI では本体以外にブロック型の支柱を開発し、機械的なコネクタで接続する仕様にした。しかし、その支柱の組み立てが利用者の組み立てにかかる負担の大部分を占めていたため、uTXII においては展示会場のブース設営などの際に用いられる組み立てが容易な市販の支柱を改良する形で uTXII を垂直に立てることを可能にした。支柱の長さをブロックのように自由に変更できないという制約が生じるが、誰もが uTXII の組み立てや分解を容易できるということを優先するためにこのような仕様とした。

### 3. 実装

#### 3.1 ハードウェア構成

図 7 に uTXII のシステム構成を示す。各側面に設置した赤外線センサは、他の uTXII と接続しているか、接続している場合にはどの uTXII のどこの側面と接続しているかを検知する。uTXI ではバネ機構を持ったシリアルインタフェースを各側面に配置し、それらを物理的に接続させることで接続の検知を行った。しかし、組み立て、分解を繰り返すうちにシリアルインタフェースに uTXI の自重による負荷がかかり、劣化しやすくなるという問題点があったため、uTXII では、赤外線センサを利用した。赤外線出力強度と出力方向を調整することで隣接する uTXII の接続の検知を実現している。内部に装着した 3 次元センサは、uTXII の 3 次元での傾きを検知する。小型で、軽量、安価に実現するために 2 次元加速度計 (ADXL202, Analog Devices) を異なる軸方向に 2 台設置することで実現している。マイクロコントローラ (H8/3664F 16 MHz, Hitachi) は、上記のセンサの出力データを処理し、1 秒ごとにそれぞれの値をメインコンピュータに送信する。また、RFID リーダ (13.56 MHz, Sofel) を内部

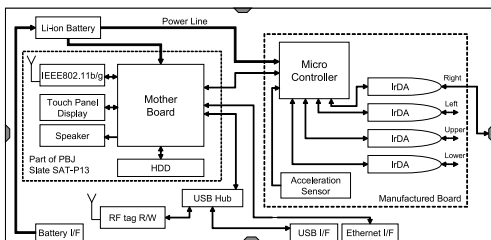


図 7 uTXII のシステム構成  
Fig. 7 Block diagram of uTXII.

に実装し、ディスプレイの外枠裏側部分に RFID アンテナを設置することでディスプレイ上に置かれた RFID タグの ID を読み取る。メインコンピュータには CPU (Pentium M 1.3 GHz, Intel) を搭載し、他の uTXII や情報機器とのデータの送受信はすべて無線ネットワーク (IEEE 802.11b/g) で行う。タッチパネルディスプレイによって利用者からの入力を検知し、スピーカとともに画像出力も行う。また、バッテリーの駆動時間は約 5 時間である。上記のメインコンピュータからバッテリーまでのハードウェアについては、ディスプレイの大きさに対してできるだけ重量を軽くすることを目的として、市販のタブレット PC (Slate SAT-P13, PBJ) から必要な機能を取り出して適用する形で実装した。

uTXII の表裏、上下、左右の定義を図 3 に示す。ソフトウェアによる組み立て形状の自己認識を上記の単純なセンサで実現するために組み立てには 3 つの制約を設けている。まず、uTXII どうしは垂直および水平にのみ接続可能である。次に、水平方向に接続する場合には、接続される uTXII の表裏面は同一の場合のみ可能である。最後に、uTXII の上側は別の uTXII の下側にのみ接続可能であり、左側は右側にのみ接続可能である。

#### 3.2 ソフトウェア構成

ソフトウェアモジュールの構成を図 8 に示す。イベント共有モジュール (Event Sharing Module)、組み立て形状認識モジュール (Structure Recognition Module)、アプリケーション起動モジュール (Application Launcher Module) の 3 つのモジュールがあらかじめインストールされている。アプリケーションモジュール (Application Module) はアプリケーションを実行するためのソフトウェアであり、それぞれにメタデータとしてアプリケーション記述データ (Ap-

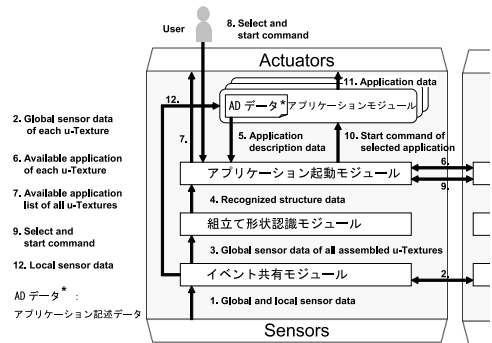


図 8 ソフトウェア構成  
Fig. 8 Block diagram of software module.

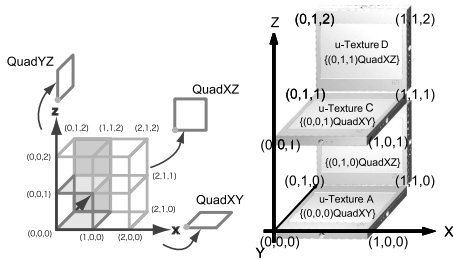


図 9 座標定義と組み立て形状の一例に対する座標

Fig. 9 Coordinates definition and coordinates of shelf-shaped object.

plication Description Data (以下, AD データ)) が存在し, アプリケーション実行可能な組み立て形状が記述される.

3.2.1 組み立て形状の自己認識

イベント共有モジュールは, 接続検知および傾き検知のセンサを含むタッチパネルや RFID リーダなどすべてのセンサデータを収集する. そして, 組み立て形状の自己認識に利用する接続と傾きに関するセンサデータは, 接続した他の uTXII と無線ネットワークを介して送受信され, 組み立て形状認識モジュールに送られる. また, それ以外のセンサデータはアプリケーションモジュールに送られ, アプリケーションの入力として利用される. uTXII に新たなセンサを設置した場合には, このモジュールを修正することでデータを取り込むことができる. 組み立て形状認識モジュールは, 入力として受け取った自己および接続した uTXII からの接続および傾きに関するセンサデータを処理することで組み立て形状と自己の位置, 傾きを出力する.

図 9 に組み立て形状に対する座標定義 (左図) と 4 台の uTXII を組み立てた一例の座標値 (右図) を示す. 4 つ角の座標値と傾き方向の値によって各 uTXII の組み立て位置が認識され, すべての uTXII の組み立て位置情報によって全体の組み立て形状が認識される. 各 uTXII では, X, Y, Z 方向の傾きの値と, 4 つの角それぞれについての X, Y, Z 座標に関する位置の値が算出され, すべての uTXII に関するそれらの値を処理することによって組み立て形状が表記される. 4 つの角の値については, 左下の角を基準として縦横 1 辺分で X, Y, Z 座標方向に数値が 1 増加する. 前述の uTXII 組み立てにおける 3 つの制約により, 同一の組み立て形状の場合には一意に座標の値を算出することが可能である. それぞれの uTXII の接続と傾きに関する値を処理することで, 組み立て形状の中で最も左下の位置に存在する uTXII が検知される. そして, その uTXII の左下の角を基点 (0, 0, 0) として全

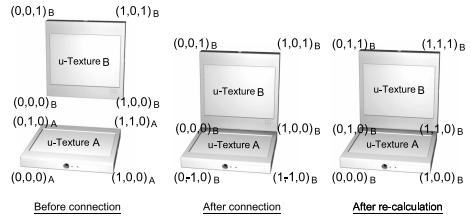


図 10 接続による組み立て形状の座標変換

Fig. 10 Example of re-calculation of uTXII B.

体形状の座標値を一意に算出する. 図 10 に組み立て形状の座標変換の例を示す. まず, 単体として存在している場合, すべての uTXII の左下の角の座標はつねに (0, 0, 0) である (図 10, 左図). そして, どこかの面が他の uTXII と接続されると, 各々の uTXII は各々の左下の角の座標を基点に処理を開始する. 図 10 の中央図は, uTXII B での座標変換前の組み立て形状の座標値である. そして, その後, 座標値の中にマイナス値が存在した場合には, そのマイナス値がゼロの値になるように全体の座標値をプラスの方向にシフトする (図 10, 右図). これらの処理によって, 同一の組み立て形状であれば, つねに同じ座標値として算出することができる.

3.2.2 アプリケーション候補の抽出

アプリケーション起動モジュールは, 検知された組み立て形状と個々の AD データを比較する. そして, SA モードの場合には, 実行可能なアプリケーションを抽出してリストとして提示し, AS モードの場合には, あらかじめ選択されたアプリケーションを実現するための形状に組み立てるための手順を提示する. AD データは, アプリケーションを実行可能な形状の記述を以下の 3 つのパラメータで表現し, 実際に組み立てられた形状の中に組み立て記述に合致する部分が含まれているかどうかでアプリケーション候補としての抽出の可否を判断する.

- (1) 組み立てられている uTXII の数の合計 (何台以上であれば実行可能か?)
- (2) uTXII の接続形態 (水平面のみか? 水平面のみで縦横同数か? 水平面と垂直面が混在するのか?)
- (3) 組み立て形状に含まれる uTXII の面方向 (X, Y, Z 面方向について, どの方向に uTXII が 1 台以上存在するか?)

図 11 に AD データの文書型定義 (DTD) を示す. AD データ中の number, form, inclination に上記 3 つのパラメータ (1), (2), (3) の値を記述する. 図 12 は, AD データの例である. この記述の場合,

```

<!ELEMENT application (path, executable-condition)>
<!ATTLIST application name CDATA #REQUIRED>
<!ATTLIST application id CDATA #REQUIRED>

<!ELEMENT path (#PCDATA)>
<!ELEMENT executable-condition(number, form?, inclination?)>

<!ELEMENT number (#PCDATA)>
<!ELEMENT form (square | cross | sequential)>
<!ELEMENT inclination (QuadXY | QuadXZ | QuadYZ)>

```

図 11 AD データの文書型定義 (DTD)

Fig.11 Document type definition of AD Data.

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE application SYSTEM "structure.dtd">

<application name="AwareShelf" id="012876">
  <path>jp.ac.keio.utexture.AwareShelf</path>
  <executable-condition>
    <number>2</number>
    <form>cross</form>
    <inclination>QuadXY</inclination>
  </executable-condition>
</application>

```

図 12 AD データの例

Fig.12 AD Data example.

uTXII の数の合計は 2 台以上で、uTXII の接続形態は水平面と垂直面が混在し、uTXII の面方向に Z 方向が含まれる、ということ表現している。つまり、図 10 の右図のような形状が含まれている組み立て形状は、すべてこの AD データに合致した形状であると判断される。図 5-3 のような組み立て形状もその条件に合致するため、アプリケーション実行可能な組み立て形状であると認識される。つまり、アプリケーション候補抽出のためにすべての組み立て形状を 1 つずつ記述することはせず、緩やかに候補を抽出する表現記述になっている。これによって、たとえば、縦横 2 台ずつ水平方向に接続した uTXII と縦横 5 台ずつ水平方向に接続した uTXII のようにスケールは異なっていながら形は同一な組み立て形状においては、1 つの組み立て形状記述によって表現できるようなスケールビリティを持たせることができる。たとえ同一組み立て形状であったとしても、実行したいアプリケーションは利用者、また、利用される状況によって異なるため、このような緩やかな抽出方法とし、最終的に 1 つのアプリケーションを選択して実行する判断は利用者が行えるようにしている。そして、最終的にあるアプリケーションが利用者を選択された場合には、ネットワークを介してアプリケーションモジュールとそれに対応した AD データが接続された別の uTXII にも自動的にコピーされ、実行される。

### 3.2.3 連携によるアプリケーションの実行

あるアプリケーションが選択され、それが実行可能な形状に uTXII が組み立てられた段階で、組み立てられた uTXII は連携してアプリケーションを実行する。まず、利用者がインタラクションしていた uTXII のアプリケーション起動モジュールから他の uTXII のアプリケーション起動モジュールに対してスタート・コマンドを配信する。そして、各アプリケーション起動モジュールは選択されたアプリケーションモジュールを各 uTXII 内で実行させる。

利用者が自分で uTXII を組み立てることを想定した場合、新しい用途で uTXII を利用したいと考えたときに新しいアプリケーションが必要になると考えられ、その際には、利用者自身が新しいアプリケーションを作成できることが望ましい。そのため、アプリケーションモジュールは、利用者が自由に作成できることを考慮した設計になっている。組み立て形状の変化や入力デバイスからの入力をイベントとした記述になっており、それらのイベントに対応した処理を記述することで新しいアプリケーションを作成することが可能である。また、将来的には uTXII が外部のネットワークに接続できる状況において、それぞれの利用者が自分で作成したアプリケーションモジュールをネットワーク上で公開したり、共有したりすることで、uTXII の利用用途を拡張できるようになると考えている。

## 4. アプリケーション例

現在までに 13 台の uTXII を実装し、有効性を確認するために 15 種類のアプリケーションを開発した。代表的なアプリケーション例を紹介する。テーブルや棚、壁として存在する uTXII が、周囲の状況や利用者の必要に応じて組み立て形状に対応したアプリケーションを提供できることが利点である。また、設定のための入力などなしに、利用者が uTXII の組み立て形状を組み替えるだけで、組み替えられた形状に対応した別のアプリケーションを実行できるようになることも利点である。

### 4.1 協調作業支援

Collaboration Table は水平方向に接続された uTXII で動作し、複数の利用者による机上での協調作業を支援するアプリケーションである (図 13)。接続した uTXII 間でドラッグ・アンド・ドロップで直接的なデータの移動が可能であり、たとえば、制作途中のデザイン画を複数の uTXII にコピーし、それぞれの利用者が個々の uTXII で独自にそのデザイン画を完成させ、最後に 1 つの uTXII に集めて比較したり



図 13 協調作業支援  
Fig. 13 Collaborative table.

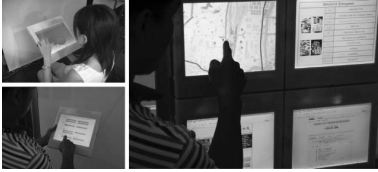


図 14 インพุットデータの融合とマルチアウトプット  
Fig. 14 Blend board.

修正したりするような協調作業を実現する。

#### 4.2 インพุットデータの融合とマルチアウトプット

Blend Board は複数の uTXII への入力を統合し、出力するアプリケーションであり、利用者の個々のアイデアや、ある質問に対する個々の回答を統合する際の利用を想定している。図 14 は複数の利用者の入力によるレストラン検索の様子を示している。この場合、それぞれの利用者は個々の uTXII で希望のレストランのジャンルについての質問に回答し、垂直方向に平面状に接続する。それに対し、接続された uTXII は、個々の入力データを統合し、その統合されたデータを基にレストラン検索を実施して、すべての uTXII に同一の候補リストを表示する。その後、利用者が候補の 1 つを選択すると、選択されたレストランに関する地図やメニュー、連絡先、店内の写真など異なったデータが個々の uTXII にそれぞれ表示される。uTXII 間の連携により表示の分担は自律的に決定される。

#### 4.3 メディアデータへの接続や携帯端末との連携

Media Shelf は、図 15 のように uTXII を水平、垂直方向に接続した棚型に組み立てることで動作可能なアプリケーションであり、棚に置かれたものを認識して動作する。それぞれのものには RFID タグが取り付けられ、識別される。たとえば、音楽 CD や映画の DVD、本を一番上の棚に置くだけで、利用者は置いたものに対応した音楽や映画、テキストコンテンツを再生、閲覧することができる。この場合、再生や閲覧の操作は、ものが置かれた uTXII に接続した垂直方向の uTXII で行う（図 15、左図）。下の棚にものを置いた場合には、置いたものの情報やコンテンツがその棚に保存され、接続した複数の uTXII で共有される。また、棚に置かれた携帯デバイスや情報機器と連携す

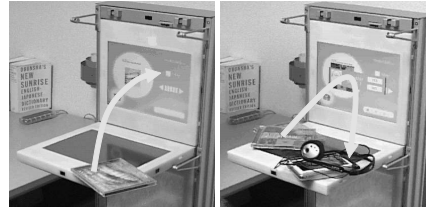


図 15 メディアデータへの接続や携帯端末との連携  
Fig. 15 Media shelf.



図 16 周囲の情報機器との連携  
Fig. 16 Appliance controller.

ることも可能である。たとえば同じ棚に音楽 CD と携帯デバイスを置くことで、uTXII はその携帯デバイス（Zaurus SL-C3000, Sharp）と連携し、音楽 CD に収録された音楽データをネットワーク経由で携帯デバイスにコピーすることができる（図 15、右図）。通常の棚の機能に加え、利用者が通常行う「ものを置く」という動作によって電子的な支援ができることが利点である。また、Media Shelf と同じ組み立て形状において、ある棚に置かれている「もの」がどの uTXII 上に置かれているかを uTXII どうしが連携して探し出すといった、「もの」の検索を行うアプリケーションも実現している。

#### 4.4 周囲の情報機器との連携

Appliance Controller は、uTXII によって周囲環境に存在する情報機器の制御や連携を可能にするアプリケーションである。たとえば、部屋の鍵に取り付けられた RFID タグを Appliance Controller アプリケーションが動作した uTXII にタッチすることで、その部屋に存在する情報機器をネットワーク経由で制御することが可能である（図 16、上図）。この場合、鍵は部屋を開錠する物理的な鍵、RFID タグはその部屋の情報機器の制御権を得る電子的な鍵として利用することができる。また、複数の uTXII を水平に接続し、ある uTXII に存在するデータを周囲の情報機器を制御している別の uTXII にドラッグアンド・ドロップすることで、その情報機器のインพุットデータとして利用することもできる。たとえば、プロジェクタを起動



し、プレゼンテーションデータを出力するような操作に利用可能である(図16, 下図)。

## 5. 議 論

### 5.1 ユーザ経験

筆者らは、国内外の学会やシンポジウムにおいて uTXII を利用してデモンストレーションを実施した。また、筆者らが所属する研究室内の公共空間や協働空間、ある家庭に一定の期間設置し、観察による知見や利用者からのコメントを得た。室内公共空間は、大学の学生や教員、関係する企業や機関の方々が自由に入りし、議論やミーティング、休憩をする空間であり、リビングルームをイメージしたレイアウトになっている。そのため、コンピュータの利用経験があまりないような方々や幅広い年齢層の方々が uTXII を利用した。それに対し、協働空間は学生や教員がものを制作したり実験を行う作業空間であり、コンピュータの経験が豊富な 20 歳前後の方々が主に uTXII を利用した。コメントを得た利用者は 19 歳から 52 歳の男女 24 名であり、コンピュータの利用経験があまりない 11 名も含まれている。

#### 5.1.1 ブロック型インタフェース

組み立て形状の接続や傾きを意識したり、それらの情報を uTXII に入力することなしに、すべての利用者が、uTXII をブロックのように各々で組み立て、アプリケーションを実行させることができた。そして、uTXII を組み立てる際に電氣的接続やその設定について意識しなかったという意見が 87.5% (21 名) であった。組み立ての機械的構造についても、容易に組み立て、分解を行うことができたというコメントが 70.8% (17 名) であった。この結果から、単体の情報機器を利用する場合と比較し、多くの利用者が認知的負荷を増加させることなく複数の uTXII を連携させることができたことを確認した。また、観察の結果、コンピュータの利用経験の有無に関わらず、利用者は uTXII を組み立て、所望のアプリケーションを実行することができていた。サーバを設置する必要もなく、設定のためにキーボードやマウスで入力を行う必要もなく、誰もが自分自身のみで uTXII を連携して利用させることができるように行った設計が妥当であったことを確認した。年齢や性別に関わらず、コンピュータの利用経験が少ない方々から特に良い評価をいただいた。一方、否定的な意見としては、uTXII の上下左右を意識して組み立てないと接続できない組み立て制約について、形状を組み替える際にそれを意識しなければ組み立てられないことが面倒であったという意見

があり、その制約を緩和して自由度を上げて欲しいという指摘があった。自由度を上げることで、たとえば向かい合った利用者が uTXII の上部どうしを接続し、連携させて協調作業を行うことが可能になる。また、uTXII の利用を観察し、コメントを得ることによって、ディスプレイ外枠裏面部分に設置している RFID アンテナについて、組み立て形状によっては RFID タグの検知に時間を要することが分かった。状況を調査したところ、Media Shelf のように uTXII が連続して垂直に接続しているような特定の組み立て配置の場合、電波の干渉によって受信感度が低下することが判明した。アンテナの大きさや配置を変更するなどの対策を検討中である。

#### 5.1.2 設置場所による利用形態

どの環境に設置した場合でも、最初の 1 週間程度は利用者の多くが uTXII に関心を持ち、組み立てや分解を試し、機能の変化を確かめていた。そして、その後は設置された環境によって使われ方が異なり、公共空間や協働空間では uTXII は、複数の利用者による協調作業に利用され、組み立てによる機能の変更が頻繁に行われた。一例として、ある企画のアイデア出しの段階でのミーティングにおいて、利用者がそれぞれの uTXII に自分のアイデアを図や文字で描き、その後、uTXII を並べてアイデアを比較したり、組み合わせたりすることに利用された。一方、家庭内においては、uTXII は頻繁に組み替えられることはほとんどなく、ある形状に一度組み立てられると、その形状で使われ続けることがほとんどであった。RFID の付いた CD を置くことで音楽を聴いたり、デジカメを置くことで撮影画像を閲覧したりというようなメディアデータや他の情報機器のインタフェースシステムとしての使われ方が好まれた。デジカメについては、利用者が RFID タグを添付したデジカメを uTXII に置くことで、その uTXII がデジカメのネットワーク上のアドレスとマシン名を検知する。そして、uTXII がそのネットワークアドレスに接続してデジカメ内のメモリにアクセスすることで、そこに保存された撮影画像を閲覧できるように実装した。

#### 5.1.3 2つのインタラクションモードの比較

2つのインタラクションモードについては、利用する際の利用者の準備状況や uTXII に関する理解状況によって、利用のされ方が異なった。AS モードは、実行したい特定のアプリケーションが決まっている利用者や、uTXII の仕組みをまだ把握していない利用者によく使われた。一方、SA モードは、組み立てたい特定の形状が決まっている利用者や、組み立て形状に対

してどのアプリケーションが実行可能かをすでに知っている利用者に使われることが多かった。また、作業開始から途中までは AS モードを使い、途中から SA モードに切り替えるような利用者も見受けられた。2つのインタラクティブモードに共通する利点の1つは、利用者が uTXII 以外のマニュアルなどを読む必要がなく、uTXII 自体からユビキタスコンピューティング環境構築のための動的な支援を受けられる点である。アプリケーションの種類を増やし、それを利用者間で共用できるようにするために、現在専用のサーバを開発している。これによって、組み立てられた uTXII の形状に対応して利用可能なアプリケーションの候補がサーバの中から自動的に検索される。そして、利用者がその1つを選択することによってネットワーク経由でダウンロードされたアプリケーションを実行することができる。uTXII を利用することによる有用性とその拡張性を広げることができると考えている。

## 5.2 今後の課題

### 5.2.1 適応領域

uTXII の適応領域としては、第1にオフィスの会議室や学校などでの利用を想定している。これらの環境では、テーブルの配置を変えたり、壁のホワイトボードに各々がアイデアを書いたり、プレゼンテーション資料を皆で閲覧したりといったように、話し合いや協調作業の形態によって空間の利用の仕方が変化するためである。そのため、そのような場合には、参加者が自由にユビキタスコンピューティング環境を組み替えられる必要がある。実際、ユーザ観察においても、上記のような形態において uTXII が利用されることが多く見受けられた。利用者による組み立てや拡張が可能な家具は数多く存在するため、uTXII は家庭内での知的な組み立て家具としての利用にも有用であると考える。家庭内で利用される情報機器が増加することが予想されるが、uTXII はそれらの種類の異なる情報機器どうしを連携させるインタフェースシステムとしての役割を担うことも可能である。

### 5.2.2 アプリケーション候補の抽出

uTXII では、スケールは異なっていながら形は同一な複数の組み立て形状においては、1つの組み立て形状記述によって表現できるよう緩やかに候補を抽出する表現記述を定義した。今後は、利用者が必要としているアプリケーション候補を適切に抽出できるようにするために、限られたパラメータで3次元的な組み立て形状をさらに正確に表現し、スケーラビリティを持たせられるようにする必要がある。組み立て形状に対して実行されたアプリケーションの履歴を保存するこ

とも考えている。それによって各組み立て形状に対して抽出するアプリケーションに優先度を付けることができる。利用者の ID を登録した RFID タグを組み立ての際に uTXII に認識させて個人ごとに履歴を保存することも考えられるが、どちらも履歴のネットワーク上での管理について考慮する必要があり、今後の課題である。

### 5.2.3 異なるタイプのブロック型情報機器の可能性

本論文では、パネル形状の uTXII を開発したが、異なる大きさや形、素材のブロック型情報機器を開発することで、利用者はそれらの特徴に対応した異なるユビキタスコンピューティング環境を実現できるようになる。たとえば、強度の強いタイル形状で uTXII と同様の組み立て形状認識機能を持ったシステムを開発すれば、それを床や天井として利用することで利用者の位置に応じた作業支援をすることが可能になる。周囲環境と調和のとれた色や形などのデザインも考慮することも重要である。今回の開発で得た知見を総合し、次のインタフェースシステム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。

## 6. 関連研究

ブロックを組み立てることでコンピューティング環境を変更したり、拡張したりすることを目的とした研究はいくつかあるが、その多くは、組み立てられたブロックの3次元形状をネットワークで接続された別のコンピュータで認識する研究や、ブロック組み立てを入力とした3次元モデリングを可能にする研究である。Algo-Block<sup>8)</sup> はセンサやマイクロコンピュータを内蔵したアルミ製のキューブであり、組み立てることでプログラミングを行うシステムである。ブロック自体からの出力は LED などである。また、Triangle<sup>9)</sup> は、手のひらに載る程度の大きさの三角形平面のブロックで、組み立てることでプログラミングを可能にするシステムである。3つの側面には電気的接合部があり、接続することでデータをやりとりし、組み立て形状を認識することができる。これらのシステムは、組み立て形状の表示やアプリケーションの出力はネットワークで接続されたディスプレイを利用する必要があり、入力としての組み立てと出力としての表示の関係が間接的である。DataFiles<sup>10)</sup> は、RFID タグが埋め込まれた透明のタイルをインタフェースデバイスとし、水平に設置された平面ディスプレイ上に配置することでコンピュータや情報機器を制御するシステムである。複数のタイルの物理的な配置によって機能合成を実現でき、映像編集や対話的なプログラミング、家電の制

御などを行うことが可能である。DataTiles のタイルは目的によって 5 つの種類に分類されており、利用者はタイルの種類を選び、組み合わせることで機能合成を行う。本研究で提案した MUI は利用者が接続方法を記憶したり、連携方法を認識したりせずに複数の機器を連携させることが可能である。uTXII では、組み立てられた時点でセンサ情報をやりとりして形状認識をし、連携のためのインタラクションを uTXII 自体が自律的に行えるようにすることで MUI を実現している。別のコンピュータやモニタなしに連携させることができ、利用者が uTXII の組合せを意識する必要がなく、利用者の組み立てに適應してそれぞれの uTXII が機能を変化させることができることが特徴の 1 つである。

また、接続することで利用者の認知的負荷を増加させることなく複数の情報機器を連携させることを目的とした研究の例としては、テーブルの天板にディスプレイが埋め込まれた ConnecTable<sup>11)</sup> がある。ConnecTable は、それらを複数物理的につなげることで各々のディスプレイを共有することができるシステムである。Multiple tablet computer system<sup>12)</sup> は、加速度計が組み込まれ、4 つの側面にタッチセンサがついたタブレット PC である。2 台を衝突させることで互いが同一の加速度変化を検知してディスプレイが共用化される。両システムともに物理的につなげたり離したりすることに対応して、個々のディスプレイがパーソナル、パブリックに切り替わることが特徴であり、協調作業に利用することを目的にしている。これらの研究は、あらかじめ実行しているアプリケーション上で 2 次元的な接続を検知し、それによってテーブルやタブレット PC が連携することを可能にしている。実行するアプリケーションを利用者があらかじめ選択したうえで複数の情報機器を連携させるこれらのシステムに対し、MUI を適用した uTXII では、組み立て形状に対して実行可能なアプリケーション候補を自律的、動的に抽出したり、あるアプリケーション実行のための組み立て形状までインタラクティブにナビゲートしたりすることができる。そのため、利用者は接続方法や連携方法を記憶、認識することなしに様々な連携を行うことが可能である。

## 7. ま と め

本論文では、利用者の認知的負荷を増加させることなく複数の情報機器を連携させることが可能なユーザインタフェースである MUI を提案し、その MUI を実現するシステムの 1 つとしてブロック型情報機器

uTXII について報告した。uTXII は組み立てられた形状を自律的に認識し、形状に対して機能を変化させ、連携動作することが可能であり、利用者は内部の計算機やセンサの存在を意識せずに組み立てることができる。そのため、利用者の認知的負荷を増加させることなく、複数の情報機器を連携させ、ユビキタスコンピューティング環境を構築、拡張することが可能になった。今後は、ソフトウェア機能の改良、機能拡張を継続すると同時に、新しい適用分野に対する検討や、別のタイプのブロック型情報機器の設計を進めていく予定である。様々な情報機器に MUI を適用することによって、多くの利用者が自分自身でユビキタスコンピューティング環境を構築、拡張できるようになると考えている。

謝辞 本研究の一部は、総務省委託研究 Ubila プロジェクトの一環として実施した。本研究の実施にあたり、協力をいただいた株式会社内田洋行に感謝する。

## 参 考 文 献

- 1) Tokuda, H., Takashio, K., Nakazawa, J., Matsumiya, K., Ito, M. and Saito, M.: SF2: Smart Furniture for Creating Ubiquitous Applications, *International Workshop on Cyberspace Technologies and Societies (IWCTS2004)*, pp.423-429 (2004).
- 2) Streitz, N., Tandler, P., Muller-Tomfelde, C. and Konomi, S.: Roomware: Towards the Next Generation of Human-Computer Interaction based on an Integrated Design of Real and Virtual Worlds, *HCI in the New Millenium*, pp.553-578 (2001).
- 3) Kidd, C.D., Orr, R., Abowd, G.D., Atkeson, C.G., Essa, I.A., MacIntyre, B., Mynatt, E.D., Starner, T. and Newstetter, W.: The Aware-Home: A Living Laboratory for Ubiquitous Computing Research, *Cooperative Buildings*, pp.191-198 (1999).
- 4) Weiser, M.: The computer for the twenty-first century, *Scientific American*, pp.94-104 (1991).
- 5) Kohtake, N., Rekimoto, J. and Anzai, Y.: InfoPoint: A Device that Provides a Uniform User Interface to Allow Appliances to Work Together over a Network, *Personal and Ubiquitous Computing Journal*, Vol.5, No.4, pp.379-388 (2001).
- 6) Kohtake, N., Iwamoto, T., Suzuki, G., Aoki, S., Takashio, K. and Tokuda, H.: State-Snap: A Snapshot-based Interface for State-Reproducible Operation of Networked Appliances, *2nd Annual International Confer-*

*ence on Mobile and Ubiquitous Systems (MobiQuitous2005)*, pp.443–453 (2005).

- 7) Kohtake, N., Ohsawa, R., Yonezawa, T., Matsukura, Y., Iwai, M., Takashio, K. and Tokuda, H.: u-Texture: Self-organizable Universal Panels for Creating Smart Surroundings, *7th International Conference on Ubiquitous Computing (UbiComp2005)*, pp.19–36 (2005).
- 8) Suzuki, H. and Kato, H.: Interaction-level support for collaborative learning: AlgoBlock an open programming language, *1st international conference on Computer support for collaborative learning (CSCL'95)*, pp.349–355 (1995).
- 9) Gorbet, M.G., Orth, M. and Ishii, H.: Triangles: Tangible Interface for Manipulation and Exploration of Digital Information Topography, *The SIGCHI conference on Human Factors in Computing Systems (CHI'98)*, pp.49–56 (1998).
- 10) Rekimoto, J., Ullmer, B. and Oba, H.: DataTiles: a Modular Platform for Mixed Physical and Graphical Interactions, *The SIGCHI Conference on Human Factors in Computing Systems (CHI'01)*, pp.269–276 (2001).
- 11) Tandler, P., Prante, T., Muller-Tomfelde, C., Streitz, N.A. and Steinmetz, R.: ConneCTables: Dynamic Coupling of Displays for the Flexible Creation of Shared Workspaces, *Annual ACM Symposium on User Interface Software and Technology (UIST'01)*, pp.11–20 (2001).
- 12) Hinckley, K.: Distributed and Local Sensing Techniques for Face-to-face Collaboration, *5th international conference on Multimodal interfaces (ICMI'03)*, Vol.36.7, pp.81–84 (2003).

(平成 18 年 6 月 22 日受付)

(平成 18 年 12 月 7 日採録)



神武 直彦 (正会員)

1998 年慶應義塾大学大学院理工学研究科修士課程修了。同年宇宙開発事業団入社。H-IIA ロケット搭載機器の研究開発に従事。2005 年慶應義塾大学大学院政策・メディア研究科後期博士課程修了。同研究科助手を経て、現在、同大学 SFC 研究所上席所員および宇宙航空研究開発機構主任開発員。ユビキタスシステム、宇宙機システム研究に従事。博士(政策・メディア)。IEEE, 人工知能学会, ヒューマンインタフェース学会各会員。



大澤 亮

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。データマイニング, ユビキタスコンピューティングの研究に従事。



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。



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、慶應義塾大学環境情報学部助教授。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。博士(工学)。日本ソフトウェア科学会, ACM, IEEE 各会員。



徳田 英幸 (正会員)

慶應義塾大学より工学修士。1983 年ウォータールー大学 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。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。