



ウェアラブルコンピューティングのためのセンサ管理デバイスの設計と実装

村尾, 和哉
竹川, 佳成
寺田, 努
西尾, 章治郎

(Citation)

情報処理学会論文誌, 49(9):3327-3339

(Issue Date)

2008-09-15

(Resource Type)

journal article

(Version)

Version of Record

(URL)

<https://hdl.handle.net/20.500.14094/90002113>



ウェアラブルコンピューティングのための センサ管理デバイスの設計と実装

村 尾 和 哉^{†1} 竹 川 佳 成^{†2}
寺 田 努^{†3} 西 尾 章 治 郎^{†1}

近年、コンピュータの小型化、高性能化にともない、ユーザがコンピュータをつねに身に着けて持ち運ぶウェアラブルコンピューティングに対する注目が高まっている。ウェアラブルコンピューティング環境では、ユーザが装着している各種センサ（ウェアラブルセンサ）を用いて、さまざまなアプリケーションが動作している。ウェアラブルシステムでは消費電力の低減が重要であるが、従来のウェアラブルシステムはウェアラブルセンサの電源を柔軟に管理できなかったため、使用していないセンサに対しても無駄な電力を供給していた。また、センサは故障や電力供給不足、過電流などさまざまな原因で動作が不安定になるが、これらの検出をソフトウェアレベルで行うことは困難であった。そこで本研究ではこれらの問題を解決するために、状況に応じた電源管理とセンシングデータ管理を実現するセンサ管理デバイス CLAD (Cross-Linkage for Assembled Devices) の設計と実装を行う。CLAD は状況に応じてセンサの電源を管理し、センサの特性を考慮した高度なエラー制御機能や故障したセンサの代わりに擬似データを生成する機能を持つ。CLAD を利用することで消費電力を低減させながらデータの正確性およびシステムの動作信頼性が向上する。

Design and Implementation of a Sensor Management Device for Wearable Computing

KAZUYA MURAO,^{†1} YOSHINARI TAKEGAWA,^{†2}
TSUTOMU TERADA^{†3} and SHOJIRO NISHIO^{†1}

In wearable computing environments, a wearable computer runs various applications with various sensors (wearable sensors). Since conventional wearable systems do not manage the power supply flexibly, they consume excess power resource for unused sensors. Additionally, sensors frequently become unstable by several reasons such as breakdown of sensors. It is inadequate for application engineers to detect them only by sensing data. To solve these problems, we propose a new sensor management device CLAD (Cross-Linkage for Assembled

Devices) that has various functions for power management and sensed data management. CLAD improves power saving, data accuracy, and operational reliability.

1. はじめに

近年、マイクロエレクトロニクス技術の発展によるコンピュータの小型化や軽量化により、コンピュータを常時身に着けて生活するウェアラブルコンピューティングに注目が集まっている。さまざまなセンサやコンピュータを身に着けるウェアラブルコンピューティングは、従来のコンピュータの利用形態と比較して次の3つの特徴を持つ^{1),2)}。

- ハンズフリー：コンピュータを身体に装着しているため、両手を使用せずに情報を参照できる。
- 常時電源 ON：コンピュータはつねに電源が入っており、使いたいときにすぐに使える。
- 生活密着：つねにコンピュータを装着した状態で日常生活を行う。

このようなウェアラブル環境では、コンピュータはユーザが装着している各種センサ（ウェアラブルセンサ）から得られたデータ（センシングデータ）を用いてユーザの行動や状況を認識することで、ナビゲーションシステム³⁾、健康管理システム⁴⁾、農業支援システム⁵⁾といったさまざまなサービスを実現している。

ユーザは表1に示すような複数のサービスを同時に利用するために多数のセンサを同時に装着する^{6)–8)}。しかし、「残存電力が少ないときは健康管理システムを使わない」「屋内に入ったときはナビゲーションシステムを使わない」といったように、ユーザの状況やデバイスの状態に応じて使わないセンサが現れる。従来のウェアラブルシステムはセンサの電源を柔軟に管理できなかったため、使用していないセンサに対しても無駄に電力を供給していた。また、センサは故障や電力供給不足、過電流などさまざまな原因で動作が不安定になるが、センシングデータだけから異常動作を検出することは難しい。

そこで本研究では、これらの問題を解決するために、ウェアラブル環境のためのセンサ管

†1 大阪大学大学院情報科学研究科

Graduate School of Information Science and Technology, Osaka University

†2 神戸大学自然科学系先端融合研究環

Organization of Advanced Science and Technology, Kobe University

†3 神戸大学大学院工学研究科

Graduate School of Engineering, Kobe University

表 1 センサを用いたウェアラブルシステム
Table 1 Wearable systems using sensors.

サービス	利用するセンサ
健康管理システム	温度センサ, GSR センサ 3 軸加速度センサ, 光電脈波センサ
ナビゲーションシステム	GPS, 地磁気センサ, ジャイロセンサ
農業支援システム	ウェブカメラ, GPS, 地磁気センサ 気温センサ, 土壌温度・水温センサ 照度センサ, 湿度センサ, 超音波センサ 赤外線センサ, 紫外線センサ
行動認識システム	加速度センサ, ジャイロセンサ

理デバイス CLAD (Cross-Linkage for Assembled Devices) の設計と実装を行う。ウェアラブルセンサとウェアラブルコンピュータの中継機器である CLAD は、状況に応じたセンサの電源制御により省電力を実現する。また、センサの特性を考慮した高度なエラー検出や、使えなくなったセンサのための擬似データ生成機能を持つ。CLAD を利用することでデータの正確性およびシステムの動作信頼性が向上する。

以降、2 章でセンサを用いたウェアラブルシステムを紹介し、3 章でシステムの設計について述べる。4 章で実装について述べ、5 章で評価を行い、最後に 6 章で本研究のまとめを行う。

2. ウェアラブルセンサを利用したシステム

ウェアラブルセンサを用いたシステムを構築する場合、一般に複数のセンサを組み合わせる場合が多い。この利用形態は、「同種のセンサを複数個使用し精度を向上させる」、「異種センサを組み合わせる精度を向上させる」、「異種センサを複数種使用しさまざまなサービスを提供する」の 3 つに分類される。以降、表 1 に示すアプリケーションを例に詳細を説明する。

2.1 精度向上のための同種センサの複数利用

精度向上のために同種のセンサを複数個使用する例としては行動認識システムがある。行動認識システムは、「歩いている」、「走っている」、「座っている」といったユーザの行動を認識し、状況に応じた情報提示⁹⁾、装着者の行動の記録や検索およびスケジュールとの連携¹⁰⁾、装着者の治療支援や早期異常検出などの健康管理に関する支援⁴⁾などのサービスを実現するための基礎となるシステムである。ユーザの行動を認識する手法としてはカメラに

よるマーカのトラッキングやセンサ情報からの学習を行うものなどが多数提案されている。なかでも、ウェアラブルセンサを用いたものとしては、認識精度を上げるために同種のセンサを複数使うエーアイキューブ社の InterTrax2¹¹⁾ や Laerhoven らの Porcupine⁷⁾ がある。

2.2 精度向上のための異種センサの利用

精度向上のために異なるセンサを複数個使用する例としては、現在位置の取得システムであるポイントマン DRM⁸⁾ があげられる。ポイントマン DRM は地磁気センサで方角を検知し、加速度センサによる歩行検知で距離を算出する。また内蔵 GPS で慣性航行の結果を補完することで高精度に位置を検出する。この例以外でも電波を使った位置測定や、RFID の読み取りによる大まかな位置検出などさまざまなセンサを用いた位置測位手法が提案されており、これらを補完的に用いることで位置精度が向上する。

2.3 さまざまなサービスのための異種センサの利用

さまざまなサービスを提供するために異なるセンサを複数利用するアプリケーションとして健康管理システム⁴⁾ や農作業支援システム⁵⁾ がある。

健康管理システムはユーザの生活習慣（食事、運動）の状況をウェアラブルセンサを用いてリアルタイムに取得し、これに合わせて生活習慣改善のためのアドバイスを提示する。具体的には腕時計型のセンサを用いて脈波、皮膚温などの生体情報と加速度データを計測する。計測データから食事中、歩行中、パソコンでキーボードを使って作業中などの動作を認識し、運動不足のときは運動不足解消を促すなど、状況に合わせてリアルタイムにアドバイスを提示する。

農作業支援システム CyferGear⁵⁾ は体の各部に装着した多種のウェアラブルセンサから気温や湿度、土壌の状態、果実の発育状況、画像、ユーザの位置などさまざまな情報を取得し、作物の発育状況の把握や、収集した情報に基づく知識獲得を行うなど、農作業を支援するさまざまなサービスを提供する。農作業ではユーザごとに必要な情報は異なり、装着するセンサも異なるためこのシステムではセンサをとりつけた手袋を複数用意し手袋を付け替えることでセンサを取りかえられるようにしている。このように、ウェアラブルシステムでは多数のセンサを取り扱いつつ、利用するセンサを自由に変更できる仕組みが必要となる。

3. CLAD の設計

ウェアラブル環境ではナビゲーションを行いながら、状況に合わせて HMD に情報を提示したり、高齢者が農作業支援システムを利用して農作業を行いながら同時に健康管理を行ったりするといったように、2 章で紹介したサービスを複数利用することが多い。ウェアラブル

ル環境ではコンピュータの電源は常時 ON であるが、たとえば、農作業支援システムは農作業時にのみ利用するが、健康管理システムはつねに利用したいというようにつねにすべてのサービスを利用しているわけではない。しかし従来システムでは、利用していないサービスのためのセンサも常時 ON になり無駄な電力を消費する。また、単一のサービス内でも、たとえば前章で述べたポイントマン DRM⁸⁾ では、GPS・加速度センサ・ジャイロセンサを利用しているが、「残存電力が十分にあるときは3つのセンサを使う」「屋内など GPS が位置情報を取得できないときは加速度センサとジャイロセンサを使用する」「残存電力が少なくなってきた場合、GPS だけ稼働させる」といったように状況に合わせて電源を管理することで、消費電力を削減できる。

さらに、センサが故障した場合、その故障を検知できないとシステムは送られてきた誤値を使い続けることになるが、ソフトウェアレベルでセンサの故障を検知することは難しい。

これらの問題を解決するには、センサ以外にセンサの状態を把握し、電源管理や故障対応を行うデバイスが必要である。そこで本研究では、センサとコンピュータの間でセンサを管理するデバイスである CLAD を提案する。以降、CLAD の設計について詳しく述べる。

3.1 システム構成

図 1 に CLAD のシステム構成を示す。CLAD は内部に専用の電源と CPU を備え、CLAD や CLAD に接続しているセンサへの電源供給を制御する。また CLAD は、電源の電圧・電流を監視し、過電圧・過電流を検出する。本論文で想定するセンサは、一般の GPS デバイスのように、PC からのコマンドに対応するためのマイクロコンピュータを搭載し、センサの種類、精度、出力範囲、起動時間、動作電圧、動作電流に関するプロファイル情報を保持している。

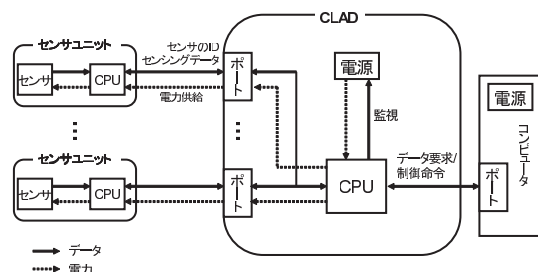


図 1 CLAD のシステム構成
Fig. 1 System structure of CLAD.

3.2 CLAD の特徴と機能

表 2 に CLAD の機能を示し、機能の実現に貢献した特徴に○印を示した。CLAD の機能は大きく電源管理とデータ管理に分けられる。以下、それぞれについて詳細に説明する。

3.2.1 電源管理

省電力

CLAD は内部の専用電源電圧を監視している。アプリケーションが、稼働しているすべてのセンサを必要としている場合は、電源残量が少なくてもすべてのセンサを稼働させる必要があるが、複数アプリケーションを利用している場合にはアプリケーション間の優劣があり、単一アプリケーション内でもセンサ間に優劣がある。これらの情報を用いることで、優先度の高いアプリケーションやセンサを稼働させつつ、消費電力を削減できる。本論文では以下に示すように、重要度・消費電力・希少度・精度・起動時間を基準としてあげる。

- 重要度：あらかじめセンサに与えられた重要度を用いてセンサを選択する。たとえば健康管理システムを利用している際、加速度センサより体温や心拍、血圧センサが重要であると考え、残存電力が少なくなれば脈拍センサなどを残す。
- 消費電力：消費電力の大きなセンサを優先的に切る。
- 希少度：数的に冗長化されているセンサから切る。たとえば複数の加速度センサと1つの距離センサを利用している場合、加速度センサは複数あるため稼働センサ数が減少しても精度は低下する恐れはあるがサービス提供は続けられる。また、後述の擬似データを用いることで、切断したセンサの値を補完することも可能である。
- 精度：速度センサと傾きセンサのように精度や分解能の異なる複数の同種センサを使用している場合、精度の低いセンサを切ることでサービスの質を大きく落とすことなく消費電力を削減する。

表 2 CLAD の特徴
Table 2 Features of CLAD

機能 特徴	電源管理			データ管理	
	省電力	代替機器	過電流検出	エラー検出	擬似データ
電源制御	○	○	○		
プロファイル情報の保持	○	○		○	○
電圧監視	○			○	○
電流監視			○	○	
センサの電圧監視				○	
センサの電流監視			○	○	
センシングデータ管理				○	○

- 起動時間：起動にかかる時間の短いセンサを優先的に切る。たとえば GPS のように起動にかかる時間の長いものを頻繁に切ると、必要となったときに時間がかかる。

これらのうち、どの基準で判断するかはユーザの意図や稼働しているアプリケーションによって異なるため、自由に指定できるようになっている。

代替機器の検索および切替え

CLAD はプロファイル情報で定義されたデータ範囲外のセンシングデータの連続受信や、センサとの通信不可などセンサの異常を検出した場合にセンサが故障していると判断し、異常センサと同種のセンサをプロファイル情報から検索し、代替機器として使用する。

過電流検出

CLAD は電圧同様、CLAD 内部の電流量を監視している。過電流は回路の故障や発火、漏電などにつながり危険である。過電流を検出した CLAD は、コンピュータに「過電流」コマンドを送信し CLAD 全体の電力供給を停止する。

3.2.2 データ管理

エラー検出

CLAD からの呼びかけに対してセンサが応答しない場合や、センシングデータの値が本来出力されるべき値の範囲を逸脱し続けている場合などの異常を CLAD が検出した場合、「電源を切断する」「代替機器に切り替える」「擬似データを生成する」「エラーの内容をコンピュータに通知する」などの対応をとる。複数の対応方法がある場合、あらかじめアプリケーションで指定された順に従う。また、エラー情報を受信したアプリケーションが独自にエラー処理することもできる。たとえば、専用電源の供給電圧低下のエラー情報を受けたコンピュータはユーザに電源の交換を促すメッセージを提示するといった対応を行える。

擬似データの生成

センサの故障やセンサに電力を供給できないといった理由でセンサを稼働させることができない場合に、アプリケーションによってはセンシングデータが入力されないと機能しないものがある。特に行動認識システムではセンシングデータに不正なデータが混入すると認識精度は著しく低下する。一般的に、あるデータに対してすべての項目や成分が埋まっているデータを完全ケース、一部にデータの抜けがあるデータを不完全ケース、すべての項目が抜けているデータを入手不可能ケースという。センサの故障などによりセンシングデータの一部が得られない状況は、不完全ケースに該当する。この不完全ケースの対処法として 3 つ考えられる¹²⁾。

- Listwise deletion
1 つでも欠損を含むサンプルは利用しない。
- Pairwise deletion
欠損部分のみを取り除いてサンプルを利用する。
- Imputation
欠損部分を何らかの値で埋めてサンプルを利用する。

センサ故障によるデータの欠損は連続的に起こるため、Listwise deletion では、認識するデータがなくなってしまう。また、Pairwise deletion においても、対処後のデータの次元数が変化するため、アプリケーションに余分な設定をする必要がある。一方、何らかの値を埋め込めば、すべてのデータを利用でき、次元数も変化しない。そこで CLAD は正常時のデータをもとに、故障したセンサの代わりに擬似的にセンシングデータを生成する機能を持つ。この擬似データ生成機能により、アプリケーションの稼働を維持でき、システム全体の動作信頼性が向上する。本論文では CLAD のエラー検出機能と従来データマイニングなどで利用されてきたデータ補完アルゴリズムを連動させることで擬似データ生成機能を実現する。

図 2 に擬似データの生成手続きを示す。以下で、センサ 1～5 の 5 個のセンサを用いて行動認識を行っている状況で、センサ 5 が故障した場合を仮定して説明する。

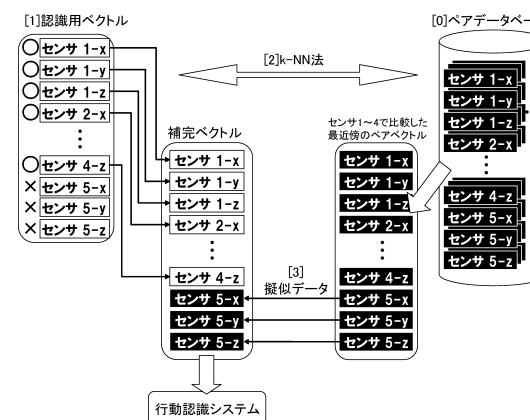


図 2 擬似データの生成方法

Fig. 2 The procedure of pseudo-data generation.

[0] ペアデータベースの構築

あらかじめすべてのコンテキストについてすべてのセンサが正常に動作しているときのデータを収集しておく。これら 1 つ 1 つのデータをペアベクトル、および構築するデータベースをペアデータベースと呼ぶ。本論文では、コンテキストを「歩く」や「立つ」などの人間の動作と定義する。

[1] センシングデータの取得

ある時刻においてユーザが装着するセンサからデータを取得する。このデータを認識用ベクトルと呼ぶ。

[2] ペアデータベースからのペアベクトルの抽出

認識用ベクトルとペアベクトルの間で k-NN 法 ($k = 1$) を用いて最近傍のペアベクトルを求める。このとき k-NN 法の対象となるのは稼働しているセンサ 1~4 のみである。

[3] ペアベクトルから擬似データの抽出

抽出したペアベクトルにおけるセンサ 5 のデータを擬似データとして認識用ベクトルのセンサ 5 のデータと置き換えて、補完された認識用ベクトル（補完ベクトル）を生成する。

図 3 に示す具体的な例を用いて説明する。例ではペアデータベースには 5 つの欠損のないデータが格納されている。そこに、センサの故障などにより一部が欠損した認識用ベクトル (15, 90, ?) が入力されると、センサ 1 および 2 の値を用いてユークリッド距離を計算する。計算により (20, 100, 100) のデータが最近傍であるため、補完ベクトルは (15, 90, 100) となる。なお、擬似データの生成プロセスは装着しているセンサや認識アルゴリズムに依存しないためさまざまなシステムに適用できる。

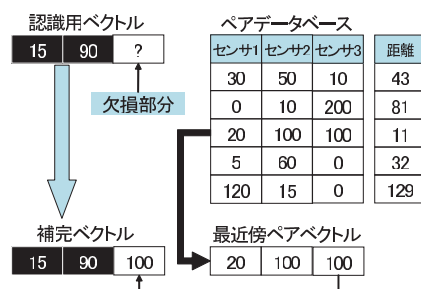


図 3 擬似データの生成の例

Fig. 3 A detailed example of the procedure for pseudo-data generation.

3.3 関連研究

CLAD に関する研究のうち、擬似データに関連するものとしては Kanagal らの研究¹³⁾があげられる。文献 13) ではカルマンフィルタや隠れマルコフモデル、パーティクルフィルタを応用することで、欠損値の補完や観測できない値の推測を行っている。しかし、カルマンフィルタを用いて予測を行うにはシステム方程式と観測方程式を立式しなければならず、直近の位置座標と現在の速度（加速度）から現在の位置を推定するように関係式が単純である場合はよいが、行動認識のように人間の体に装着した加速度センサの値を予測するには人間の骨格から方程式を立式しなければならず適用は困難である。また、仮に立式したとしてもセンサの装着位置や組合せの変更に応じて式を変更する必要があるため汎用性は低い。隠れマルコフモデルを用いた手法ではセンシングデータの振舞いからセンサが壊れているか否かの粒度の低い推測のみが行われており、そのままセンシングデータの予測に適用することは困難である。仮に可能であるとしても Baum-Welch アルゴリズムを利用してパラメータの学習を行うためには膨大な学習データが必要になるという問題もある。さらに文献 13) 中ではパーティクルフィルタを用いて時刻から温度や湿度の推測を行っているが、本研究で取り扱っている加速度センサなどのデータは他の絶対的なパラメータとの関連性が低く、時刻などから予測することは難しい。本研究ではさまざまなセンサにおいて汎用的に補完を行う機構を提供しており、新規性を有するといえる。

また、ウェアラブルコンピューティング環境における電源問題に関しては、コンピュータやセンサなどのデバイス自体を省電力化する解決方法⁷⁾がこれまで提案されているが、中間ハードウェアを用いて電源の ON/OFF を制御することで問題を解決する取り組みはこれまでにない。Phidget¹⁴⁾は加速度、温度などのセンサやモータ、LED などのアクチュエータを Phidget Interface という中継機器に自由に接続して使うツールキットであるが、Phidget はコンピュータにデータを渡すだけでデータ管理や電源管理は行っていない。

電源電圧および電流監視はハードウェアを構築するうえで従来から考慮されてきた点である。中継機器¹⁵⁾として高機能なものはエラー制御を行っており、擬似データも含めて動作信頼性向上のためには必要な機能であるため CLAD に実装した。

4. CLAD の実装

CLAD のプロトタイプの回路図を図 4 に示す。CLAD の処理装置としては Microchip 社の PIC16F873A を用い、Microchip 社の MPLAB 上で CCS 社の PIC C Compiler を用い

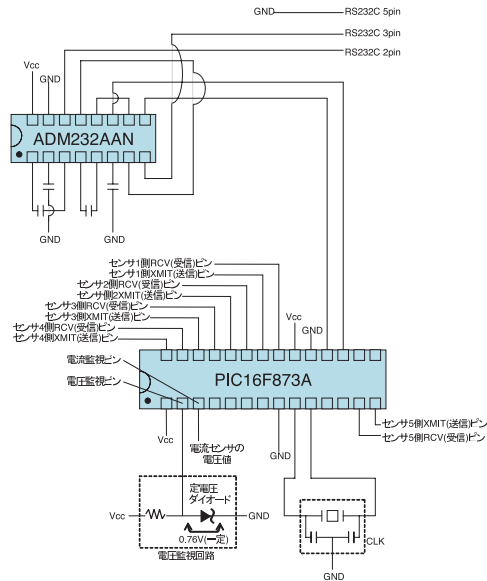


図 4 CLAD の回路図
Fig. 4 CLAD circuit.

て実装を行った。センサやウェアラブルコンピュータをプロトタイプに接続した様子を図 5 に示す。さらに、CLAD の仕様を表 3 に示す。

CLAD とコンピュータおよび CLAD とセンサ間は RS232C で通信し、通信速度は 9,600 bps とした。CLAD は 5 つのポートを持ち、最大 5 個のセンサを制御できる。実装した各ウェアラブルセンサは処理装置を持ち、CLAD との通信制御などを行う。本論文の評価実験では SONY 社の VGN-U50 (Celeron M 900 MHz) をウェアラブルコンピュータとして使用した。

4.1 通信プロトコル

CLAD とコンピュータおよび CLAD とセンサ間でやりとりするデータフォーマットを図 6 に、制御コマンドを表 4 に示す。以下、これらの通信プロトコルについて詳細に説明する。

CLAD 起動時

CLAD はすべてのポートに電力を供給する。電力供給されたセンサは、保持しているプ

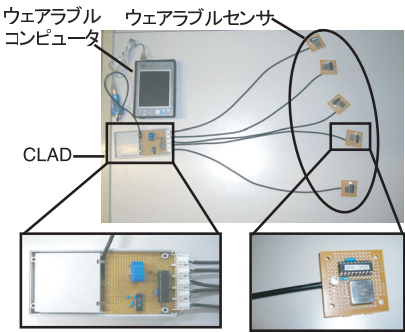


図 5 CLAD のプロトタイプ
Fig. 5 CLAD prototype.

表 3 CLAD の仕様
Table 3 CLAD specifications.

通信方式	RS232C
電源	5 V
通信速度	最大 9,600 bps
外形寸法 (本体のみ)	W 76 × H 13 × D 70 mm
重量	電池・ケーブル除く 130 g 電池・ケーブル含む 292 g
消費電力 (CLAD 単体)	0.05 W

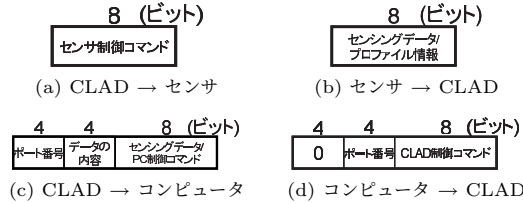


図 6 データフォーマット
Fig. 6 Data format.

ロファイル情報を CLAD に送信する。次に、CLAD はコンピュータに「起動」コマンドを送信し、コンピュータとのコネクションを確立し、センサのプロファイル情報をコンピュータに送信する。

表 4 コマンド一覧
Table 4 Command tables.

(a) センサ制御コマンド		(b) PC 制御コマンド	
コマンド		コマンド	
電源オフ センシングデータ要求 プロフィール情報要求 電源オン		センサ	過電圧
			過電流
			データ異常
			センサ異常
		CLAD	起動
終了			

(c) CLAD 制御コマンド	
コマンド	
擬似データ生成	あり
	なし
代替機器利用	あり
	なし
重要度変更	高
	低
取捨選択基準変更	重要度
	希少度
	消費電力
	精度
	起動時間
電力供給	開始
	停止
センシング	開始
	終了
プロフィール情報要求	
CLAD 終了	

表 5 CLAD と各種センサの消費電力
Table 5 Power consumption of CLAD and sensors.

名称	消費電力 [W]
イメージセンサ	1.150
GPS	0.505
距離センサ	0.250
CLAD (本体のみ)	0.050
水分センサ	0.035
ジャイロセンサ	0.025
地磁気センサ	0.010
加速度センサ	0.010
湿度センサ	0.003

に「センシングデータ要求」を送信し、受信したセンシングデータをコンピュータに送信する。その際、図 6(c) のデータの内容に「正常値」もしくは「補完値」という情報を埋める。CLAD からコンピュータへセンサ異常の通知など制御コマンドを送信する際は、PC 制御コマンドに該当センサのポート番号を付加してどのセンサの異常か分かるようにする。

新規センサ参加時

新たにセンサが接続されたとき、センサはプロファイル情報を CLAD に送信する。以降は CLAD 起動時と同様、新規センサとのコネクションを確立しプロファイル情報をコンピュータに通知する。

センサと CLAD の通信切断時

コネクタが外れるなどによってセンサと CLAD の通信が切断した場合、CLAD はセンシングデータ要求がタイムアウトすることによりセンサの切断を認識し、センサの電源を停止後「センサ異常」コマンドをコンピュータに通知する。

4.2 電源管理

CLAD と各種センサにおける消費電力の計測結果を表 5 に示す。表 5 から明らかのように CLAD の消費電力はセンサの消費電力と比べてそれほど大きくない。また、表 5 に示す消費電力は、センサ本体 1 個あたりの消費電力であるため、複数のセンサを使った場合、システム全体における CLAD が占める消費電力の割合はさらに小さくなる。

電圧監視

CLAD の電圧監視回路は図 4 に示すように抵抗と定電圧ダイオードから構成される。このとき、CLAD の電圧監視ピンで読み取る電圧値はつねに一定 (0.76 V) となる。電圧監視ピンの入力電圧を A/D 変換したものを V_{vd} とすると PIC の A/D 変換は 1,024 段階で

あるため、

$$V_{vd} = \frac{0.76 [\text{V}]}{|V_{cc} [\text{V}]|} \cdot 1024$$

が成立する。したがって、たとえば低電圧の閾値を 4 [V] と設定した場合、 V_{vd} が 0x0C3 を上回れば低電圧状態と判断する。

電流監視

電流監視には LEM 社¹⁶⁾ の電流センサ (Current Transducer LTS 6-NP) を用いる。このセンサの電圧特性として電流が流れていない状態で 2.5 V を出力し、0.3125 V/A の割合で線形増加する。電流センサの出力電圧を電流監視ピンで読み取った電圧を A/D 変換したものを V_{id} とすると PIC の A/D 変換は 1,024 段階であるため、

$$V_{id} = \frac{0.3125 \times I [\text{A}] + 2.5}{|V_{cc} [\text{V}]|} \cdot 1024$$

が成立する。したがって、たとえば過電流と判断する閾値を $I = 2 [\text{A}]$ と設定した場合、 $V_{cc} = 5 [\text{V}]$ のとき V_{id} が 0x210 を上回れば過電流と判断する。

4.3 インタフェース

図 7 に実装した CLAD とパソコンとのインタフェースである CLAD viewer を示す。CLAD viewer は PIC ベースの CLAD を PC 上から制御できるソフトウェアである。この CLAD viewer で CLAD のポートに接続されているセンサの種類などのプロファイル情報、

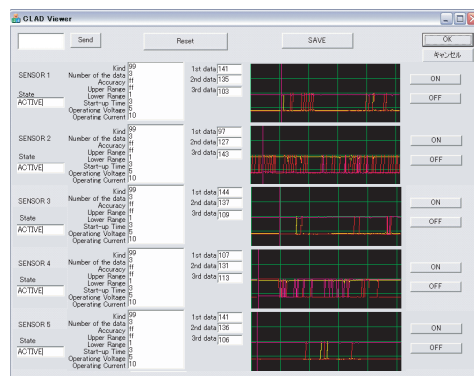


図 7 CLAD viewer
Fig. 7 The viewer for CLAD.

センサの接続状態、出力データとその時系列のグラフを確認できる。さらに各センサの電源の ON/OFF, CLAD へのコマンドの手動送信、出力データの保存ができる。

ただし、本論文では PIC のメモリの少なさから、CLAD ではペアデータベースを保持せず、PC 上にペアデータベースの構築およびデータ補完の機構を実装して評価した。しかし、最終的には CLAD に高性能なプロセッサを搭載し、その記憶容量や処理速度の範囲内で擬似データ生成を行うことで、PC 上の評価で得られた結果と同様の性能を得られると考えている。

5. 評 価

本章では CLAD の特徴である電源管理による省電力性、エラー制御による擬似データの有効性を検証する。以下実験の詳細について説明する。

5.1 省電力性

使用する加速度センサの数を変化させたときの消費電力の実測値を図 8 に示す。ここで使用する加速度センサはプロセッサを搭載した CLAD 用センサである。

図 8 中の「CLAD なし」は 5 個の加速度センサの総消費電力であり、電源制御を行わないため実際に稼働しているセンサの数が変わっても一定値のままである。一方、稼働しているセンサにのみ電力供給する CLAD は、稼働センサの数を減らすにつれ消費電力が減少している。ここで使用する加速度センサはプロセッサを搭載した CLAD 用のセンサであり、表 5 に示した加速度センサの消費電力より大きくなっている。また、稼働センサ数が 0 個の場合でもセンサは待機電力を消費する。表 5 に示す CLAD の消費電力は CLAD 単体の

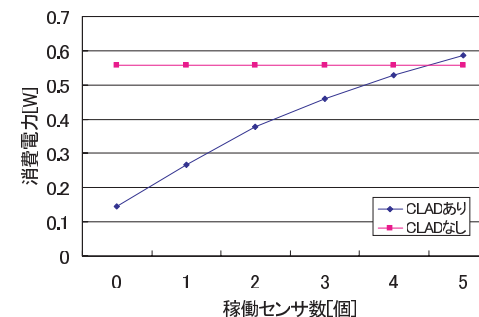


図 8 稼働センサ数と消費電力
Fig. 8 Active sensors vs. power consumption.

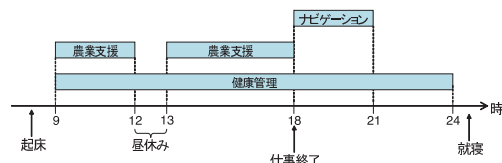


図9 ウェアラブルシステム利用者の1日

Fig. 9 A day of a wearable system user.

ものであり、センサは接続されていない。図8のCLADありの稼働センサ0個が示す値はCLAD単体とセンサ5つの待機電力の和である。稼働センサ数が4個と5個の間で両曲線が交差している。これは、CLAD本体の消費電力が影響しているためで、装着しているセンサの稼働率が約90%以下であればCLADを装着した場合の方が消費電力が少ないことが分かる。

図9にウェアラブルシステムを利用した場合の1日の行動例を示す。この行動例は図8に示した電源制御による省電力の具体例であり、アプリケーションに応じてセンサのON/OFF制御を行う想定でのシミュレーションで、得られた消費電力量は個々のセンサの消費電力から算出した理想値である。このユーザは健康管理システム、ナビゲーションシステム、農業支援システムを図9のように利用する。それぞれのシステムに必要なセンサの消費電力はおよそ0.1W、0.6W、1.7Wである。これらの消費電力のうち健康管理システムは文献4)より、ナビゲーションシステムはカメラ、GPS、各種センサを、農業支援システムはGPS、地磁気、ジャイロを装着した場合に表5より算出した概算値である。CLADを利用せず朝9時から夜24時まですべて稼働させると総消費電力は $(0.1 + 0.6 + 1.7) \times 15 = 36 \text{ Wh}$ 、CLADを利用して必要なときのみ電力を供給すると表5に示すCLADの消費電力を加算しても $\underbrace{0.1 \times 15}_{\text{health care}} + \underbrace{0.6 \times 3}_{\text{navigation}} + \underbrace{1.7 \times 8}_{\text{agriculture}} + \underbrace{0.05 \times 15}_{\text{CLAD}} = 17.65 \text{ Wh}$ となりCLADを利用した場合の消費電力削減率は約51%となる。ナビゲーションではGPSと慣性航行を使い分けるなどしてアプリケーション内で必要でないセンサがある場合に切断するとさらに消費電力を抑えられる。このようにCLADを介してセンサを装着すると消費電力が大きく削減できることが分かる。たとえば、標準的な単3二次乾電池(1,800mAh)を4本直列を2組で計8本用い、シナリオで消費する36Whおよび17.65Whを15時間で平均的に使用し、5V駆動と仮定すると、CLADなしの場合半分の7.5時間しか稼働できないが、CLADありの場合15時間以上稼働できる計算になる。

5.2 擬似データ出力の有効性

本論文では、擬似データの有効性を検証するために、行動認識システムを例に認識率を評価する。評価のために構築した行動認識システムは、スター精密株式会社製静電容量型3軸加速度センサ(ACA302)を5個用い、「立っている」、「歩いている」、「走っている」の3つのコンテキストを認識する。システムは、時刻 $t = T$ に加速度センサが生成する15次元のセンシングデータ(認識用ベクトル)の各成分 $c_{i,t}$ ($i = 1, \dots, 15$)について時刻 T から遡った20サンプル分の平均 $\mu_{i,T}$ および分散 $\sigma_{i,T}$ を求める。

$$\mu_{i,T} = \frac{1}{20} \sum_{t=T-19}^T c_{i,t}$$

$$\sigma_{i,T} = \frac{1}{20} \sum_{t=T-19}^T \left\{ c_{i,t} - \mu_{i,T} \right\}^2$$

計算量を下げるために認識用ベクトルの平均 μ および分散 σ を x 軸、 y 軸、 z 軸ごとに平均値をとり、次元数を30次元から6次元に減らしたものを X_T と定義した。また X_T の平均 M_T および分散 S_T を用い、以下の式で特徴ベクトル Z_T を定義した。

$$Z_T = \frac{X_T - M_T}{S_T}$$

認識させたい特徴ベクトル(認識用特徴ベクトル)とあらかじめ学習させたすべての特徴ベクトル(学習済み特徴ベクトル)のユークリッド距離を算出し、k-NN法($k = 1$)を用いることでコンテキストを認識する。

5.2.1 実験の手続き

前節で述べた行動認識システムを用いて認識精度を評価した。あらかじめ、被験者に両足首、両手首、腰の5カ所に加速度センサを装着してもらい「立っている」、「歩いている」、「走っている」の3つの行動を順番にとってもらい、ベアベクトルおよび学習済み特徴ベクトルを収集した。次いで、各被験者に3つの行動からなる一続きの行動を1回行ってもらい、認識率を測定した。センサの電源を切ることで擬似的にセンサが故障した環境を作り、5個のセンサがすべて故障する以外の場合(計31通り)について評価した。31通りで使用した認識用データは同一のものである。また、比較対象として表6に示す3通りの手法を用いた。なお、壊れたセンサのデータを0および128にしているのは、最小値および中央値が読み取られた場合を想定している。本評価実験における被験者は男性2名、女性1名の健康な23歳の大学生・大学院生である。

表 6 提案手法と比較対象

Table 6 Proposal and comparative methods.

提案手法	壊れたセンサの代わりに擬似データを用いて認識
比較手法 1	壊れたセンサを除外し、正常なセンサのみで認識
比較手法 2	壊れたセンサのデータをすべて 0 として認識
比較手法 3	壊れたセンサのデータをすべて 128 として認識

5.2.2 結果と考察

図 10, 図 11, 図 12 に実験結果を示す。横軸は正常なセンサと故障しているセンサの組合せ (○は稼働, 空白は故障) を示し, 縦軸は認識精度を示す。実験結果から分かるように, いずれの被験者においても提案手法が最も高い認識率となった。稼働しているセンサだけで認識を行う比較手法 1 では, センサの組合せによってはうまく認識できない。これは正常なセンサがそのコンテキストの特徴をとらえていないためである。また, 比較手法 1 や比較手法 2 では, センサ 1 個の故障が認識精度を大きく変化させる。これは本認識システムが特徴量として分散を用いているが, パディングのデータはつねに一定で変化しないため分散が 0 となるためである。

ペアデータベースのサンプル数の影響

ユーザごとに認識率が異なる理由は, 個々の動きの特徴による違いや, ペアデータベースのサンプル数の違いが考えられる。前述の 3 つのコンテキストに対し 300 個, 200 個, 100 個, 50 個, 10 個, 1 個のペアベクトルで構築したペアデータベースで認識率を評価した。試行回数は各ペアベクトル数に対して 10 回で, いずれの試行のペアベクトルも学習時に用いたデータサンプルから無作為に抽出されている。各ペアベクトル数における 10 回の試行の平均値を図 13 に, 稼働センサ数ごとの平均値を図 14 に示す。結果より, 全体的にペアベクトル数の増加に対して認識精度の向上が見られるが, ペアベクトル数間の有意差を検定するために Fisher の分散分析 (ANOVA: analysis of variance)¹⁷⁾ を行った。ペアベクトル数 i の水準の母平均を μ_i として, 仮説を立てる。帰無仮説 $H_0: \mu_{300} = \mu_{200} = \mu_{100} = \mu_{50} = \mu_{10} = \mu_1$, 対立仮説 H_1 : 少なくとも 1 つの母平均が異なる, として H_0 を正しいと仮定する。検定統計量は $F(10-1, 60-6)$ 分布に従い, 危険率 5% で検定を行った結果, 棄却域は $R = \{f > 2.386\}$, 分散分析表より分散比 $f = 1.069 < 2.386$ となり, 検定値が棄却域に入らないため H_0 を採択する。この結果, 検定有意, 危険率 5% で 6 水準間に有意差は認められなかった。一方, ペアベクトル数 300 個と 1 個の平均の差を Welch の t -検定で検定した。危険率 5% で検定を行った結果, “2 群の平均値は等しい” という帰無仮説の棄却域は $R = \{t > 2.2622\}$ とな

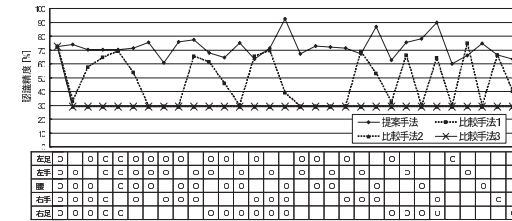


図 10 ユーザ 1 の結果

Fig. 10 The result of user 1.

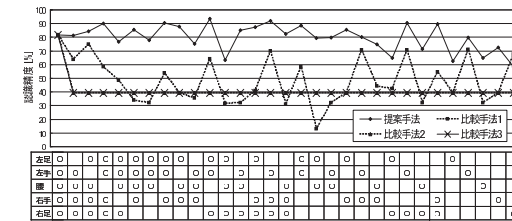


図 11 ユーザ 2 の結果

Fig. 11 The result of user 2.

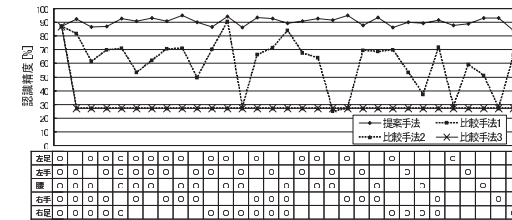


図 12 ユーザ 3 の結果

Fig. 12 The result of user 3.

り, 検定値 $t = 11.5836 > 2.2622$ であるため, 有意差が認められた。検定結果より, ユーザごとに認識率が異なる理由はペアデータベースの大きさではなく, 被験者の動きの特徴によるものであることが分かった。

また, ペアデータベースのサンプル数が 300 において補完および認識の処理時間を計測

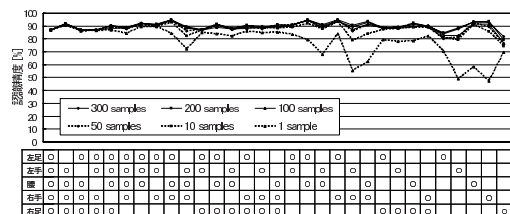


図 13 ペアデータベースのサンプル数と認識率の変化
Fig. 13 Pair-database vs. accuracy.

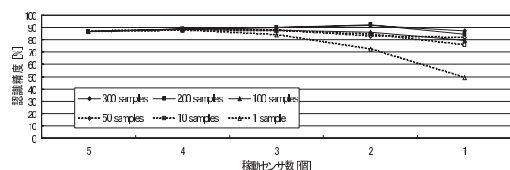


図 14 ペアデータベースのサンプル数と認識率の平均値の変化
Fig. 14 Pair-database vs. mean of accuracy.

したところ、1 サンプルあたり補完に 2.5 ミリ秒、認識に 2.7 ミリ秒、計 5.2 ミリ秒要した。

さらに、ペアデータベースの規模が一定以上大きくなると補完精度にほとんど影響を与えないため、補完精度に影響を与えない程度までペアベクトル数を減少させることで擬似データ生成の処理時間の短縮および、ペアデータベース容量の削減ができる。さらに、単純にペアベクトル数を減少させるのではなく、自己組織化マップなどの競合学習¹⁸⁾などにより効率的にペアデータベースを構築することで、ペアデータベースの規模を一定に抑えつつ、幅広いセンシングデータの補完も可能となる。

擬似データの精度

図 10 から図 12 に示すように、擬似データを用いた方がセンサの故障がない場合より認識率が高くなることが多い。これは、学習済み特徴ベクトルとペアベクトルは元は同じデータであるため、補完された認識用ベクトルはすべてのセンサが正常に動作しているときの認識用ベクトルより学習済み特徴ベクトルに近いのである。つまり、正常なセンサが多いと最近傍のペアベクトルが真のコンテキストである確率が高く、そのデータを用いて補完するため認識率がすべて正常なときより高くなる。一方、つねに擬似データを用いれば高精度の認識が可能とは限らない。たとえば 5 個のセンシングデータのうち 4 個が類似し、1 個が独

立している 2 つのコンテキストを認識する場合に独立しているセンサが壊れるとする．類似している 4 個のセンサから選ばれた最近傍のペアベクトルの独立したセンサの部分の値にばらつきが生じるため，真の値から離れる場合が多くなる．

システムリソースについて

本論文では PIC のメモリの少なさから擬似データ生成を PC 上で行ったが、PC における補完プログラムの使用するメモリはペアデータベースを含めて約 6 MB であり、十分オンメモリで動作する。また、処理中の CPU 使用率は 1~2% 程度であり、CPU が補完処理を行うことで余分に消費する電力は単純に計算しても数 mW~数十 mW である。しかも、これらのリソースはセンサ故障時のみに利用されるもので、つねに消費されるものではない。また、高性能マイコンを使うことで CLAD 内に補完処理を実装することも今後の予定として考えており、この場合でも超低消費電力版 SH マイコンなどを用いると、その利用による消費電力は 100 mW 程度となる。さらに、表 3 に示すように、擬似データ生成処理を除く CLAD の消費電力は 50 mW であり、この 50 mW はセンサのエラー検出や電流・電圧監視機能を提供するために用いられている。上記にも述べたが、センサ故障時に数十 mW の電力を用いてデータ補完を行うこととなるが、センサ未故障時にはデータ補完にかかる消費電力は必要とされない。センサ故障時には補完を行わないとシステムの精度の低下や停止を招くため、余分な消費電力を持ったとしても CLAD は有効であるといえる。

6. む す び

本研究ではウェアラブルセンサのためのセンサ管理デバイス CLAD を構築した。CLAD は、状況に応じてセンサの電源を管理することで省電力を実現する。また、センサの特性を考慮した高度なエラー制御をはじめ、センシングデータを管理することで高いデータ信頼性を実現している。評価実験より、CLAD の省電力性および、擬似データの有効性が明らかになった。

今後は、CLAD の小型・軽量化や CLAD を用いた応用アプリケーションの実装、センサの相互関係を利用した擬似データの生成による認識率の向上を行っていきたい。

また、同一センサを複数アプリケーションが利用している場合、あるアプリケーションの判断で他のアプリケーションにとって重要なセンサを制御することのないように、排他制御やアプリケーション間の調停などが必要になる。現在、本論文であげたセンサ取捨選択基準からシステム内の総合的な重要度スコアを算出するなど、より高度なセンサ制御手法を提案する。

謝辞 本研究の一部は、文部科学省科学研究費補助金基盤研究（A）（17200006）、（20240009）および特定領域研究（19024046）によるものである。ここに記して謝意を表す。

参 考 文 献

- 1) 塚本昌彦：モバイルコンピューティング，岩波新書（2000）.
- 2) Miyamae, M., Terada, T., Tsukamoto, M. and Nishio, S.: Design and Implementation of an Extensible Rule Processing System for Wearable Computing, *Proc. International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services (MobiQuitous 2004)*, pp.392–400 (2004).
- 3) Kanbara, M., Tenmoku, R., Ogawa, T., Machida, T., Koeda, M., Matsumoto, Y. and Kiyokawa, K.: Nara Palace Site Navigator: A Wearable Tour Guide System Based on Augmented Reality, *Proc. CREST/ISWC Workshop on Advanced Computing and Communicating Techniques for Wearable Information Playing*, pp.7–14 (2004).
- 4) 鈴木琢治，大内一成，土井美和子，森田千絵，佐藤 誠：LifeMinder：ウェアラブル健康管理システム，情報処理学会第 65 回全国大会，pp.219–222 (2003).
- 5) 中尾太郎，寺田 努，塚本昌彦，宮前雅一，庄司 武，岸野泰恵，義久智樹，西尾章治郎：ウェアラブル型ルールデータベルシステムを用いた農作業支援システム，情報処理学会第 65 回全国大会論文集，pp.211–214 (2003).
- 6) Laerhoven, K.V., Schmidt, A. and Gellersen, H.W.: Multi-Sensor Context Aware Clothing, *Proc. International Symposium on Wearable Computers (ISWC 2002)*, pp.49–57 (2002).
- 7) Laerhoven, K.V. and Gellersen, H.W.: Spine versus Porcupine: A Study in Distributed Wearable Activity Recognition, *Proc. International Symposium on Wearable Computers (ISWC 2004)*, pp.142–149 (2004).
- 8) シリコンセンシングシステムズジャパン株式会社：歩行者用慣性ナビユニットポイントマン DRM. <http://www.sssj.co.jp/>
- 9) Ho, J. and Intille, S.S.: Using Context-Aware Computing to Reduce the Perceived Burden of Interruptions from Mobile Devices, *Proc. Conference on Human Factors in Computing System (CHI 2005)*, pp.909–918 (2005).
- 10) 山根隼人，長尾 確：AcTree：状況認識・記録による個人行動支援，情報処理学会第 66 回全国大会論文集 (2004).
- 11) エーアイキューブ株式会社. <http://www.aicube.com/>
- 12) 浅野正彦：欠損データの分析法，選挙学会紀要，No.1, pp.101–123 (2003).
- 13) Kanagal, B. and Deshpande, A.: Online Filtering, Smoothing and Probabilistic Modeling of Streaming data, Technical report, UMD CS (2007).
- 14) Phidgets Inc. <http://www.phidgets.com/>

- 15) Husemann, D., Narayanaswami, C. and Nidd, M.: Personal Mobile Hub, *Proc. International Symposium on Wearable Computers (ISWC 2004)*, pp.85–91 (2004).
- 16) LEM. <http://www.lem.com/>
- 17) 大澤秀雄：基礎から学ぶ統計学，梓出版社 (2005).
- 18) Kohonen, T.: *Self-Organizing Maps*, Springer (1996).

(平成 19 年 10 月 29 日受付)

(平成 20 年 6 月 3 日採録)



村尾 和哉（学生会員）

2006 年大阪大学工学部電子情報エネルギー工学科卒業。2008 年同大学大学院情報科学研究科博士前期課程修了。現在，同大学院博士後期課程に在籍。ウェアラブルコンピューティング，コンテキストウェアネスの研究に興味を持つ。日本データベース学会の学生会員。



竹川 佳成（正会員）

2003 年三重大学工学部情報工学科卒業。2005 年大阪大学大学院情報科学研究科マルチメディア工学専攻修士課程修了。2007 年同専攻博士課程修了後，神戸大学自然科学系先端融合研究環重点研究部助教となり，現在に至る。博士（情報科学）。音楽情報科学，ウェアラブルコンピューティングの研究に従事。



寺田 努（正会員）

1997 年大阪大学工学部情報システム工学科卒業。1999 年同大学院工学研究科博士前期課程修了。2000 年同大学院工学研究科博士後期課程退学。同年より大阪大学サイバーメディアセンター助手。2005 年より同講師。2007 年神戸大学大学院工学研究科准教授。現在に至る。2004 年より特定非営利活動法人ウェアラブルコンピュータ研究開発機構理事，2005 年には同機構事務局長を兼務。2004 年には英国ランカスター大学客員研究員を兼務。博士（工学）。アクティブデータベース，ウェアラブルコンピューティング，ユビキタスコンピューティングの研究に従事。IEEE，電子情報通信学会，日本データベース学会，ヒューマンインタフェース学会の各会員。



西尾章治郎（正会員）

1975 年京都大学工学部数理工学科卒業。1980 年同大学院工学研究科博士後期課程修了。工学博士。京都大学工学部助手，大阪大学基礎工学部および情報処理教育センター助教授，大阪大学大学院工学研究科情報システム工学専攻教授を経て，2002 年より大阪大学大学院情報科学研究科マルチメディア工学専攻教授となり，現在に至る。2000 年より大阪大学サイバーメディアセンター長，2003 年より大阪大学大学院情報科学研究科長，その後 2007 年より大阪大学理事・副学長に就任。この間，カナダ・ウォータールー大学，ビクトリア大学客員。データベース，マルチメディアシステムの研究に従事。現在，Data & Knowledge Engineering 等の論文誌編集委員。本会理事を歴任。電子情報通信学会フェローを含め，ACM，IEEE 等 8 学会の各会員。