

論文要旨

本論文は人の骨格情報を用いた NUI (Natural User Interface) における、姿勢・動作を認識するアルゴリズムの開発に関する論文である。

UI (User Interface) は、Windows を代表とする GUI (Graphical User Interface) が主流であるが、近年では任天堂が発売している Wii をはじめ、スマートフォンにおける iPhone などのタッチパネルによる入力や音声認識など、人が直感的に操作を行う UI が増えている。

このような背景において Microsoft 社のゲームコンソール Xbox360 用 UI として発表した、「Kinect」の登場により、これまでよりも簡単に人の動作を認識できるようになった。これからさらに人の自然な動作によって、操作を行う NUI が盛んになってくるだろう。誰もが NUI による操作を行えるようにするには、個人による差を考慮しなければならない。Kinect では人の骨格情報を取得しているが、人ごとに骨格や動作速度が違うため、動作を認識しづらいといった問題点がある。また身振り手振りなどの動作「歩く」「物を動かす」などの自然な動作を入力動作として認識してしまうといった問題もある。自然動作と操作動作を区別できなければ、誤動作が多くなる。

以上に挙げた問題を解決するため、個人差に左右されない入力アルゴリズムが必要だと考え、CartooNect (カートゥネット) , GAMIC (ガミック) , AccuMotion (アキュモーション) の 3 種のデモコンテンツ、アルゴリズム、ツールの開発を行った。CartooNect では溜め動作の認識を行った。一定時間同じポーズをとるといった溜め動作を認識させることで、自然動作と入力動作を区別することに成功した。GAMIC では、姿勢認識のアルゴリズム化を行った。操作動作の骨格情報を設定しておき、リアルタイムで取得している人の骨格情報と比較し、類似度を求める評価関数を開発した。これらの評価関数を用いることで、姿勢認識を高速に行うことが可能になった。AccuMotion は先に開発した溜め動作認識、評価関数による姿勢認識を応用した操作ツールである。上下左右の入力を認識する十字キーのような操作を実現することで、ゲームだけでなく家電操作やデジタルサイネージに応用できると考えている。

今後の展望としては、AccuMotion をアプリケーション化し、デジタルサイネージやエンタテイメントシステム、家電製品の操作などに生かしたい。さらに幅広いユーザを対象による評価を行いより使いやすい NUI を研究する可能性を有している。

目 次

1. はじめに	1
1.1 ユーザインターフェースの歴史	1
1.2 Kinect とは	1
2. 研究背景	4
2.1 Kinect の開発	4
2.1.1 Kinect for Windows SDK	4
2.1.2 OpenNI と NITE	5
2.2 NUI を使った作品	7
2.2.1 Xbox360	7
2.2.2 ネットワーク動画サイトで発表された作品	7
2.2.3 MMD への応用	9
2.2.4 Team Lab Mirror	9
2.2.5 エンタテイメント・デモ「NAIST Ballpool」	10
3. NUI の課題	11
3.1 現在の NUI の問題点	11
3.1.1 幅広いユーザ	11
3.1.2 一体感のある操作	11
4. 開発：3つのケーススタディ	13
4.1 NUI を用いたシリアルゲーム「CartooNect」	13
4.1.1 CartooNect の目的	13
4.1.2 CartooNect のコンセプト	13
4.1.3 溜め動作の認識アルゴリズム	15
4.1.4 まとめ	15
4.2 没入感の高いリアルタイムアニメーションツール「GAMIC」	16
4.2.1 GAMIC のコンセプト	16
4.2.2 姿勢認識アルゴリズム	16
4.2.3 ツール「GAMIC」	18
4.2.4 まとめ	18
4.3 AccuMotion	18
4.3.1 AccuMotion 概要	18
4.3.2 AccuMotion のアルゴリズム	19
4.3.3 実験	19
5. 評価	23
5.1 CartooNect の評価	23
5.2 GAMIC の評価	23
5.3 AccuMotion の実験結果	24

1. はじめに

1.1 ユーザインターフェースの歴史

ユーザインターフェースとは、人が機械やシステムを簡単に利用するための総称である。

初期のコンピュータは、パンチカードや磁気テープ入力がデバイスであり、CRT (Cathode Ray Tube, ブラウン管) によりインタラクティビティが向上する時代になると、キーボードによる文字入力によってコマンドを入力し操作していた。このようなインターフェースはCUI (Character User Interface), CLI (Command Line Interface) と呼ばれ普及した。文字による表現は、使用者にとって理解しやすく、インターフェースを多く実装しなくてすむので、現代でもプログラム開発、ハードウェア操作などで使用されている。しかし操作するためのコマンドを、人間が記憶しておく必要があるため、キーボード操作に慣れないユーザや身体的にハンディがあるユーザにとっては使いづらいインターフェースともいえる。

1973年にXeroxのパロアルト研究所で開発された「Alto」では、直感的に操作できるGUI (Graphical User Interface) ソフトウェアが搭載された。ディスプレイに表示されるファイルやアプリケーションをグラフィカルに表示し、マウスによって操作できるインターフェースになった。コマンドを記憶する必要がなくキーボード操作や、CLIが不向きな人でも簡単に操作でき、その後のコンピュータ開発に大きく影響を与えたといえる。Macintosh, Windows OSに搭載されたことで現在広く普及している¹⁾。

近年では様々な技術を用いたインターフェースが開発・研究されている。

加藤博一らはAR技術を用いたTUI (Tangible User Interface, タンジブルユーザインターフェース) のプロトタイプシステムを開発した²⁾。

現在iPhoneでは、マルチタッチ入力が可能なタッチパネルデバイスによる操作方法を用いており、マウスを使うよりも直感的に操作できるインターフェースとなっている。また最近のiOS5では音声認識システム「Siri」が用いられており、音声によってアプリケーション操作を行うことができる(図1.1)。さらに加速度センサを用いて人の動きを認識できる。任天堂が2006年に発表したゲームプラットフォーム「Wii」でもコントローラに加速度センサを用いており、手の動きによってゲームをすることが可能になっている。

以上に挙げた音声・動作認識など、人の振る舞いをそのまま操作に生かすインターフェースをNUI (Natural User Interface) と呼ばれている。次節では、このようなセンサの最新技術である「Kinect」で、人の全身の動作を簡単に認識できるようになった技術について解説する。

1.2 Kinectとは

KinectはMicrosoft社が北米市場で2010年11月4日に発売した、Xbox360用のインターフェースである(図1.2)。KinectはRGBカメラ、赤外線プロジェクター、赤外線センサ、4つのマイクロホンを用いて、人の動きや音声を認識することができる。

Kinectは赤外線による深度センサを用いて、使用者が特別なデバイスを装着することなく骨格情報を取得することができる。赤外線を用いた深度センサはTOF (Time of Flight) 方式とLightCoding方式に分けられる。TOF方式は赤外線を照射して、戻ってくるまでの時間差によって奥行きを測定している。LightCoding方式はPrimeSense社が開発した方式で、非点

収束差法と呼ばれる CD-ROM のピックアップ部のオートフォーカス技術に近いと考えられる³⁾。赤外線プロジェクターから特徴的なパターンで赤外線を照射し、パターンの歪みを赤外線センサで撮影し奥行きを求めている(図1.3)。ゲーム用のインターフェースとして速度が足りないという理由で、時間差分を使う TOF 方式は採用されなかったのではないだろうか⁴⁾。

深度センサは、赤外線を吸収してしまう黒色の物や反射素材を利用した対象物、太陽光など強い光の環境下では照射した赤外線がうまく認識できず、正確に深度情報を取得できないというデメリットがあるが家庭用ゲーム機では問題ないだろう。

これまで人の動作を取得するには、人にマーカーをつけ複数台のカメラでの撮影、RGB 画像や赤外線画像を画像処理するモーションキャプチャーなど大規模で高額なものが多く、開発環境が限られていた。そこで安価に高精度なモーションキャプチャが行える Kinect は、発売前から注目されており、発売直後には Adafruit Industries 社が賞金を懸けてまで解析が行われた。当初 Microsoft 社側はハッカーによる Kinect の解析、開発行為に難色を示したが、後に Kinect の内部アルゴリズムへのアクセス、Xbox360 での不正行為を目的としたプログラム以外の開発を容認している。現在では Microsoft 自身において、NUI に関するセミナーや勉強会が活発に行われるようになっている⁵⁾⁶⁾。



図 1.1 iPhone4S の音声認識「Siri」



図 1.2 Microsoft 社の Kinect

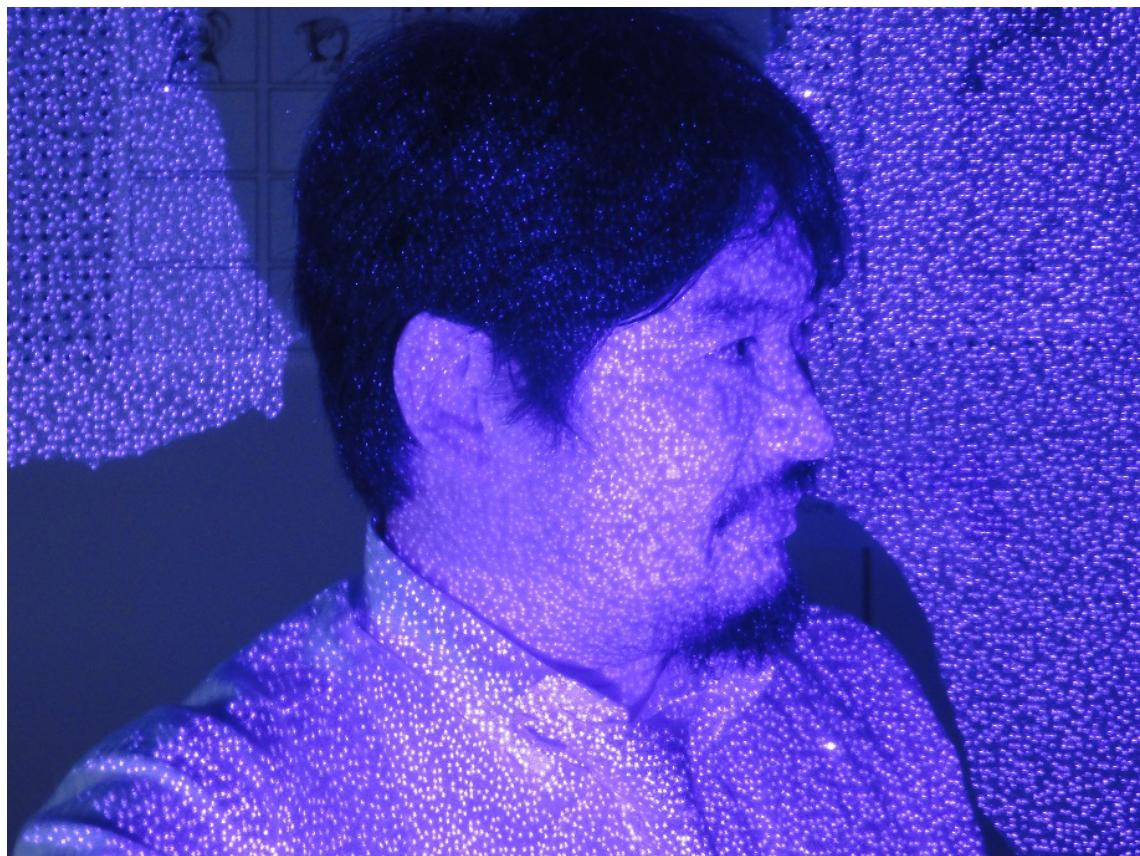


図 1.3 LightCoding のパターンを赤外線カメラで撮影した様子

2. 研究背景

2.1 Kinect の開発

2.1.1 Kinect for Windows SDK

Kinect for Windows SDK とは Microsoft 社が公式に配布している SDK (Software Development Kit) である。モーションキャプチャ機能とオーディオ機能を利用できる二つの API (Application Programming Interface) が提供されている (図 2.1)。深度センサによるユーザの認識は 6 人、骨格情報の取得は 2 人までとなっている。骨格情報は、図 2.2 に示す全身 20 か所を取得できる。

2012 年 2 月 1 日には、商用利用向けの新しい Kinect と開発パッケージを発売する予定である。新しい Kinect は Windows OS のアプリ操作などに用いることができるという⁸⁾。

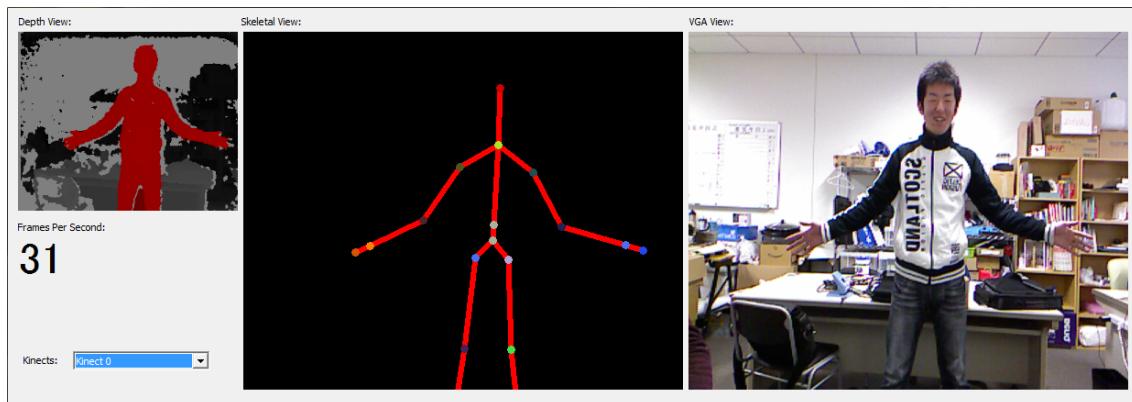


図 2.1 Kinect for Windows SDK の骨格取得サンプルプログラム

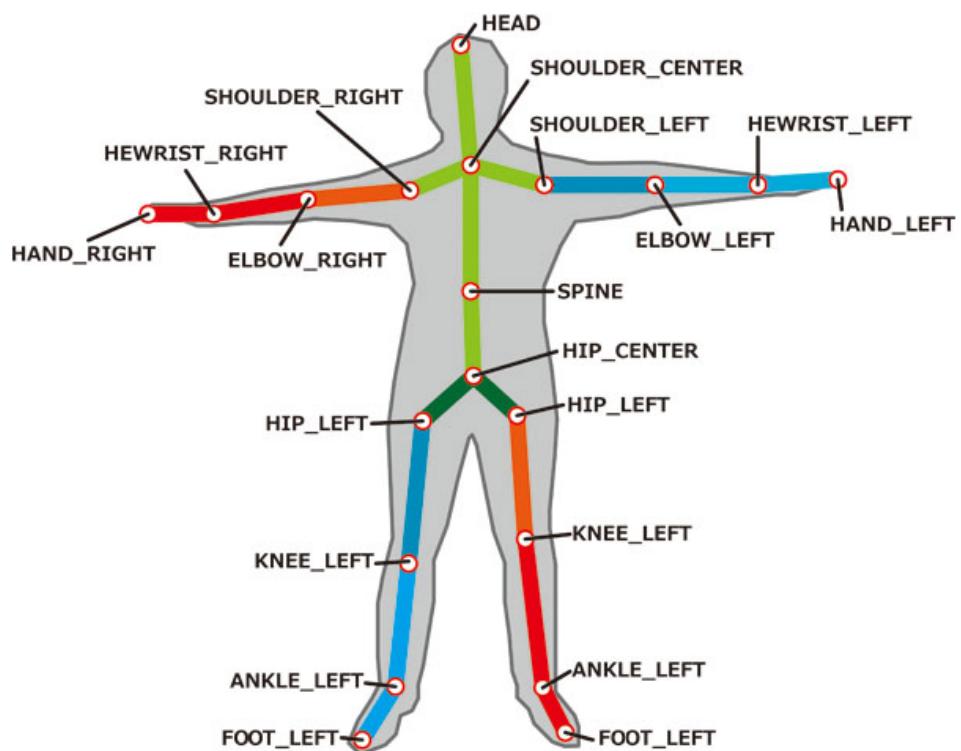


図 2.2 骨格情報の構造

2.1.2 OpenNI と NITE

OpenNI とは Kinect の開発元 , PrimeSense 社が中心となって開発をしているオープンソースの開発環境である⁹⁾ . OpenNI の NI は Natural Interaction の略で , 「人の感覚に基づいた新しいインタラクションの概念」と公式サイトのドキュメントで説明されている¹⁰⁾ . 新しいインタラクションの例として ,

- 人の音声をつかったコマンド入力
- 手のジェスチャーによる操作
- 全身動作によるゲーム

が挙げられている .

Kinect for Windows SDK にはない , 上半身のみ , 下半身のみといった体の一部分の認識が可能になっている (図 2.3) .

NITE は姿勢認識を行うミドルウェアである . OpenNI は NITE の機能を使うことによって , ユーザの検出や追跡を行う¹¹⁾ . また NITE を使うことで , 手を振る , 円を描くといった , 簡単なユーザの動作を認識できる .



図 2.3 OpenNI のサンプルプログラム「NiUserTraker」

OpenNI は Kinect だけでなく , Kinect に類似した機器を使用できる . この仕様により , OpenNI の機能を多くの機器で使用することができ , 機器に依存しない NUI の開発が可能となった .

現在 , ASUS 社が販売している「Xtion Pro」「Xtion Pro Live」(図 2.4) という機器が Kinect に次ぐ PrimeSense 社のリファレンスに基づく深度センサとして販売されている . ASUS 社は 2011 年 6 月には「Xtion Pro」を用いたコンテストを開催し (図 2.5) , 家電 , PC アプリ向け NUI システム「WAVI」の開発を精力的に行っている .



図 2.4 ASUS 社の Xtion Pro Live

A screenshot of the ASUS Xtion PRO developer challenge website. The header features the ASUS logo and navigation links for Product, Community, Event (which is highlighted in yellow), and Download. The main content area shows a collage of various icons representing different applications and technologies. On the left, there's a monitor displaying a desktop interface with icons for Video, Photos, and Internet. Below it, the Xtion PRO camera is shown with the text "Xtion PRO sensor replaces your controllers!". To the right, a section titled "Congratulations! Aliens Stole My Beer wins USD\$ 20,000!" lists the winners of the developer challenge and vote-and-win activity. The winners of the developer challenge are: First prize: Aliens Stole My Beer (USD \$20,000); Second prize: Evade (USD \$3,000); Third prize: American Sign Language Translator (USD \$1,000); Best video shooting: Evade (USD \$500). The winners of the vote-and-win activity are: (1) Eee Pad Transformer (wjfriend@xxx, ben@xxx, jg_ong@xxx); (2) WAVI (esrahosgor@xxx, jason.lange@xxx, leif@xxx, jik@xxx, mc112376@xxx); (3) O!Play Gallery (robotcowbell@xxx, eanalau@xxx, dagadagad@xxx, yuhui84@xxx, gigan22@xxx, fcheekin@xxx, mu.az@xxx, dennysaputra@xxx, tongtongyx405@xxx, nick.schellevis@xxx). A note at the bottom states that all winners will receive an email notice by the end of August, 2011. At the bottom right, there are buttons for Rules, Finalist (with a "Vote" button), and FAQ. The Xtion logo with the tagline "Intuitive living room experience" is also present.

図 2.5 XtionPro を用いたコンテスト

2.2 NUI を使った作品

2.2.1 Xbox360

Xbox360 で Kinect はゲームでアバター操作、メニュー画面操作に用いられている。

アバター操作は、取得した骨格情報をアバターのボーンに反映してリアルタイムアニメーションを行っている。さらに「Kinect アドベンチャー」内では、キャラクターアニメーションと合成してより没入感を高めている。この技術は Omek Intaractive 社が発表した論文「Enhanced interactive gaming by blending full-body tracking and gesture animation」にて報告されており、キャラクターアニメーションの開始フレームと終了フレームにキャプチャーデータをブレンディングすることで、体験者は没入感を得られると記載されている¹²⁾。

メニュー画面では、操作方法は、左右の手を動かすと画面上のポインタを移動でき、静止状態が続くと決定とみなされるものになっている（図 2.6）。また、自分のアカウントで顔を登録しておくと、自動ログインが可能になっている。



図 2.6 Xbox360 のメニュー画面

2.2.2 ネットワーク動画サイトで発表された作品

OpenNI と NITE を使ったホビー作品は、ネットワーク動画共有サイトで多く発表されている。これらはホビイストによる作品ではあるが、技術的に高度で実装速度でも大変早期に行っている。「なりきりウルトラセンブン」（2011 年 1 月 4 日、hogehoge335 氏）は Kinect 開発の初期に youtube で公開された作品である¹³⁾。画像処理によって人をウルトラマンセブンのように表現し、特定のポーズをとると対応した技のエフェクトが出るようになっている（図 2.7）。また同じ開発者が公開した「Kinect でかめはめ波」（2011 年 1 月 31 日、hogehoge335 氏）では、「ドラゴンボール」で有名な「かめはめ波」を再現して打つことができる（図 2.8）¹⁴⁾。エネルギーを溜める動作の時間によって、発射される「かめはめ波」の威力が変わってくる。



図 2.7 「なりきりウルトラセブン」の動画より



図 2.8 「Kinect でかめはめ波」の動画より

2.2.3 MMDへの応用

MMD (Miku Miku Dance) とは , 初音ミクを主とした 3D キャラクタモデルを用いてアニメーションコンテンツを作成するフリーソフトウェアである . キャラクタのアニメーション付けは手付けキーフレームによって行われていたが , Kinect の登場により人の動作をリアルタイムにキャラクタへ反映させることができるようにになった . 動作の録画 , 再生を行うことができ , 人の自然な動作をアニメーションで再現できるようになった .

右上に認識した人物が表示され , 骨格情報が取得できると配置されているキャラクタを操作できる . リアルタイムでモーションキャプチャを行う場合 , 体の一部が隠れてしまうと , 隠れた部分の骨格情報が取得できず , 対応しているキャラクタの部分があらぬ方向に向いてしまうので注意が必要となる .



図 2.9 MMD での Kinect 利用 (右上に深度画像が表示される)

2.2.4 Team Lab Mirror

Team Lab Mirror とは , TeamLab · ComputerVision チームの斎藤康毅氏が開発した画面上で洋服を試着できるシステムである¹⁵⁾ . 画面上に表示されている自分の体にシャツとズボンを重ねることで , 試着したような体験ができる . 洋服の選択は手を左右に動かすことで , 次の洋服に切り替わるようになっている . 手の胸の付近で動かすとシャツの切り替え , 腰の付近でズボンの切り替えが行える (図 2.10) .

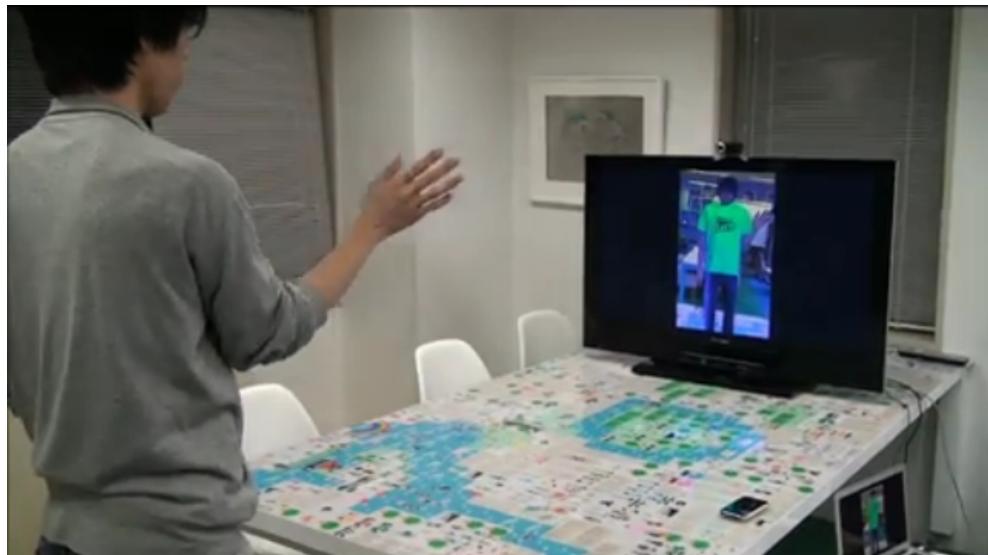


図 2.10 Team Lab Mirror

2.2.5 エンタテイメント・デモ「NAIST Ballpool」

奈良先端科学技術大学院大学が開発した、NAIST Ballpool は物理演算を使用した 2D ゲームである¹⁶⁾¹⁷⁾。プレイヤの動きをロボットのキャラクタに反映し、物理エンジンによって生成されたボールプールの中で自由に遊ぶことが出来る（図 2.11）。二人同時に認識でき、ブロックの積み上げや、ボールの投げ合いなどさまざまな運動を協力しながら体験することができる。

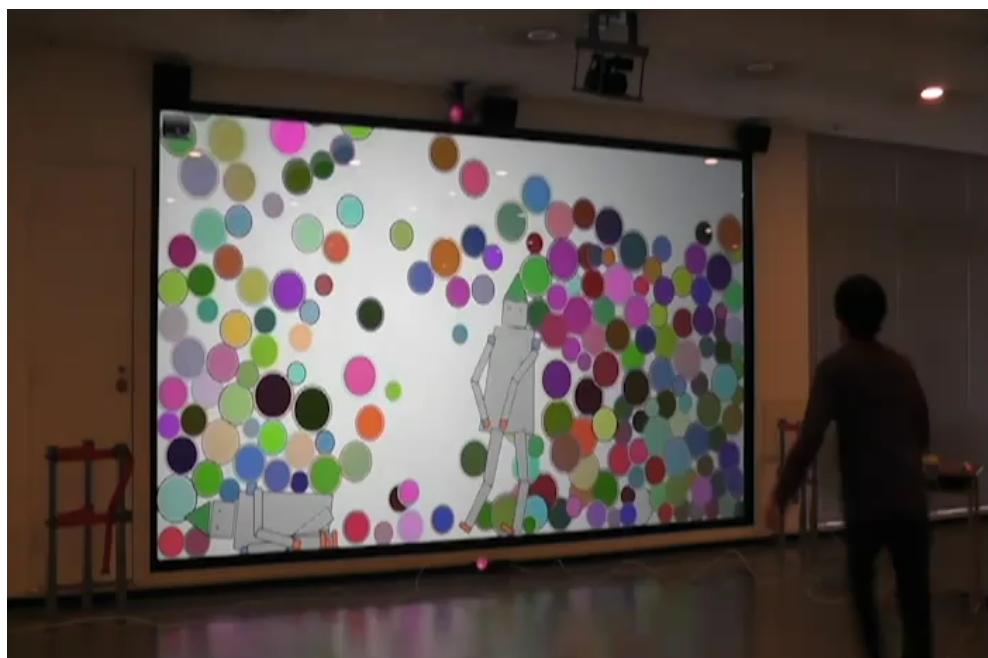


図 2.11 NAIST Ballpool

3. NUI の課題

3.1 現在の NUI の問題点

3.1.1 幅広いユーザ

2章に挙げたような作品は、NUIによる操作を行っているが「幅広いユーザでは利用できない」という課題がある。

例えば「なりきりウルトラセブン」や「Kinect でかめはめ波」は、youtube 動画上で開発者は上手く扱うことができるよう見える。しかし公開されたソースコードを追跡してみると、

- 思ったとおりに技がでない
- 溜め動作を行っても、エネルギーが溜まらない
- 関係のない動作をしていたら技がいきなり発射される

など開発者本人以外では上手く操作できない問題があることを発見した。

また、NUIを使った操作方法の例として、Microsoft 社で行われたセミナー内⁵⁾で、画面内に設定した領域に手が近づいた場合に入力判定を行う「領域方式」(図 3.1)、手の移動量を用いて上下左右の操作を行う「フリック動作法式」が紹介された。しかしこれらの方式は、三割程度の動作精度しかないというデジタルサイネージ製作者から報告された。

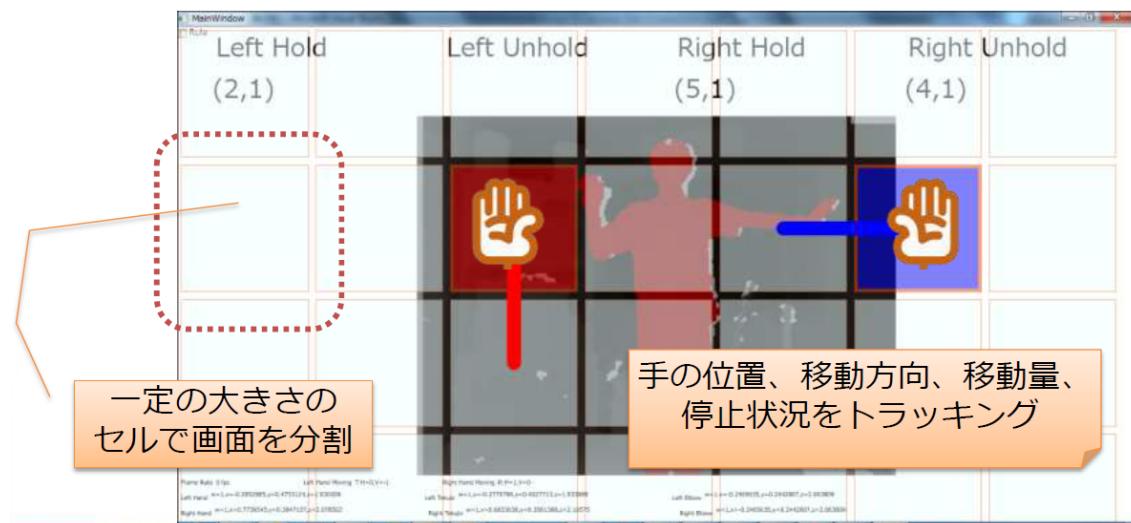


図 3.1 領域方式

3.1.2 一体感のある操作

また一体感が無いと「誇張されたアニメーションを人間が再現できない」という問題がある。ゲームシステムでは「手を振る」「パンチ」「キック」などの基本的な動作はの多くは、OmeK Interactive 社や MMD の直接ボーンに割り当てることでリアルタイムでキャラクタアニメーションに反映することができる。しかし宙返り、必殺技など誇張されたアニメーションを人間が再現することは難しい。Omek の手法では事前にアニメーションを多く登録しておく

必要があり、宙返りや必殺技など誇張されたアニメーションの再現ができないと¹²⁾において報告されている。

NUIによる操作方法の認識は、個人による差を考慮しなければならない。ユーザによって異なる骨格の長さ、動作の速さ等の違いを包括して、誰が操作動作を行っても認識できるようにならなければならない。また、歩いているときに手を振ったと誤認識されてしまうと、ユーザが意図して行った操作動作と、自然動作を区別する必要がある。

以上のような背景から、Kinectから取得した情報から安定して自然動作と操作動作の判別、操作動作の種類の認識を行う必要がある。

操作動作と自然動作の区別は、操作開始動作の段階で溜めを認識することにした。一定時間同じ動作（溜め）を行うことで、ユーザが意図して行うことを認識できる。

姿勢認識は、関節の座標などでは、体格が違った場合操作できない場合が多くなり操作にくくなってしまう。そこで個人による骨格情報の違いを包括で出来るよう、姿勢認識には各関節の角度を用いることにする。さらにアルゴリズム化を行う事で、多くの姿勢を認識できるようになると考えた（図3.2）。

この仮説を元に、プログラム開発を行った。

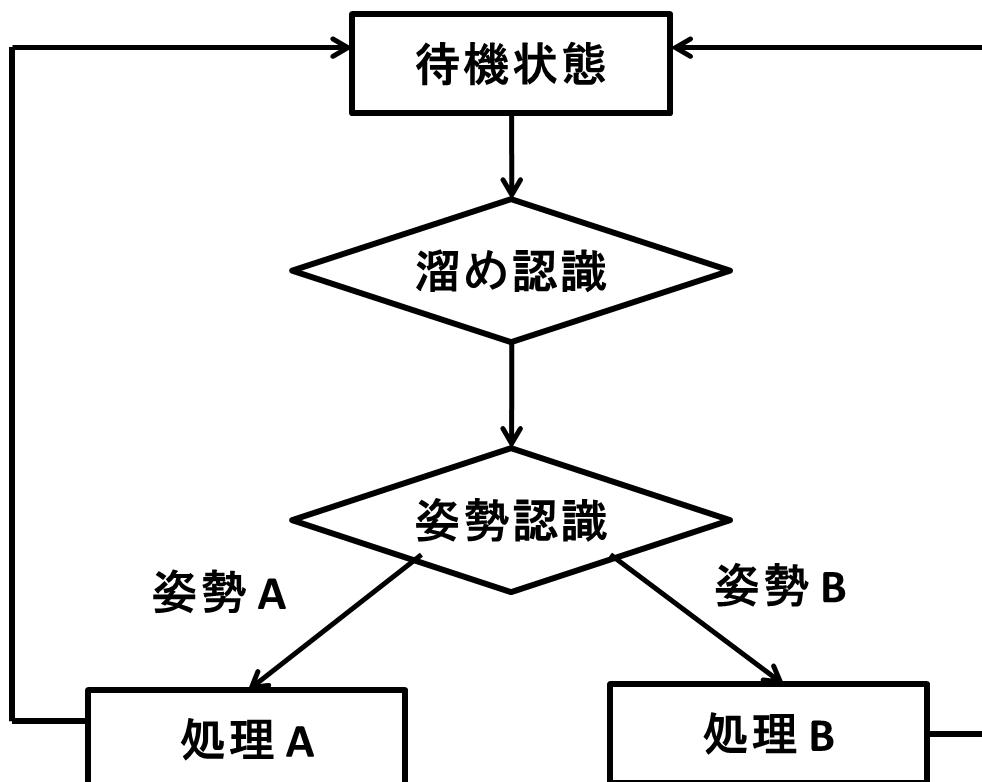


図3.2 フローチャート

4. 開発：3つのケーススタディ

4.1 NUI を用いたシリアルスゲーム「CartooNect」

4.1.1 CartooNect の目的

CartooNect はシリアルスゲームという形で、日本の災害後の農業における放射能被害と復興の可能性を表現したコンテンツである。

シリアルスゲームとは、社会的問題をゲーム化することで体験的に理解をさせる、教育的目標をもったゲームである¹⁸⁾。放射能汚染の問題は、2011年3月11日に発生した東日本大震災以降長年にわたり日本に存在するもので、農業関係者だけでなく、消費者、子供から大人、お年寄り、外国人など幅広い多くの人々がこの問題について共有することが必要だと考え、テーマとして取り上げた。

Kinect を使った全身インタラクションを用いた理由は、野外での活動による放射能汚染を恐れ、比較的安全な屋内で遊ぶことになった子供たちが激しい運動を出来るようにしたためである。リアリティの高い操作方法を実装することで、VR 技術を活用した全身運動を中心とする「感覚運動遊び」を実現した（表 4.1）¹⁹⁾。ピアジェの段階説は表 4.1 の上から順に進行していく、順序が入れ替わることはないといわれる。よって子供だけでなく大人も楽しめるシリアルスゲームとなる。日本語に限定しない幅広い体験者に「楽しい、面白い」と感じてもらうとともに、ゲーム内で体験したことを探れず、隠さず、力強い復興を共有してほしいため、全身運動によるインタラクションの実現を目的とした。

表 4.1 ピアジェの遊びの段階説

	期間	解説
感覚運動遊び	生後～2歳頃まで	感覚を獲得し、運動しているだけで楽しいと感じる
象徴的遊び	2歳～6歳頃まで	物を記号化できるようになり、ごっこ遊びなどする
ルール遊び	7歳以降	人間関係を理解できるほど思考が具体的になり、ルールのある遊びをする

4.1.2 CartooNect のコンセプト

CartooNect という名前は「Cartoon（漫画・絵）」と「Kinect」を組み合わせた造語である。体験においてユーザは、ゲーム世界での背景となる絵を画材を使って自由に描く。絵はスキヤンし、即座にゲーム世界の背景として用いる（図 4.2）。この絵には日本の復興を願った応援メッセージが書かれていたものもあった。ゲームは自分の描いた世界が放射能汚染されているという設定で、放射性物質を吸収するヒマワリを咲かせることがゲームの目的となっている（作品発表後にヒマワリの放射能吸収能力は極めて低いとの調査報告²⁰⁾があった）。

ヒマワリを咲かせたい場所へキャラクタを移動させ、プレイヤがしゃがむとキャラクタもしやがみ状態となりエネルギーを溜める。プレイヤが伸びをしたタイミングで、辺りにエネルギーを解き放ちヒマワリを成長させる。この動作を繰り返すことでヒマワリが大きく育て、たくさんのヒマワリを咲かせ大きく育て得点を競う（図 4.1）。



図 4.1 CartooNect のスクリーンショット



図 4.2 描いてもらった絵の一部

4.1.3 溜め動作の認識アルゴリズム

CartooNect は溜め動作の認識を行うため、待機状態、しゃがみ状態、伸び状態の 3 種類の状態を設けた。しゃがみ状態の認識は、プレイヤの両膝の間接が一定以上の角度で曲げており、両手が両膝より下にある時認識する。伸び状態の認識は、両手が頭の上にきており、両膝の関節が伸びている時認識する。

待機状態のまま伸び状態を認識しないように、しゃがみ状態を経由しなければ伸び状態を認識しないように状態遷移するようにした（図 4.3）。しゃがみ状態が検知されており、他の動作や伸び状態を認識していない場合、プレイヤーは溜め動作を行っていると認識できる。こうすることで、一連の動作から溜めを認識することに成功した。

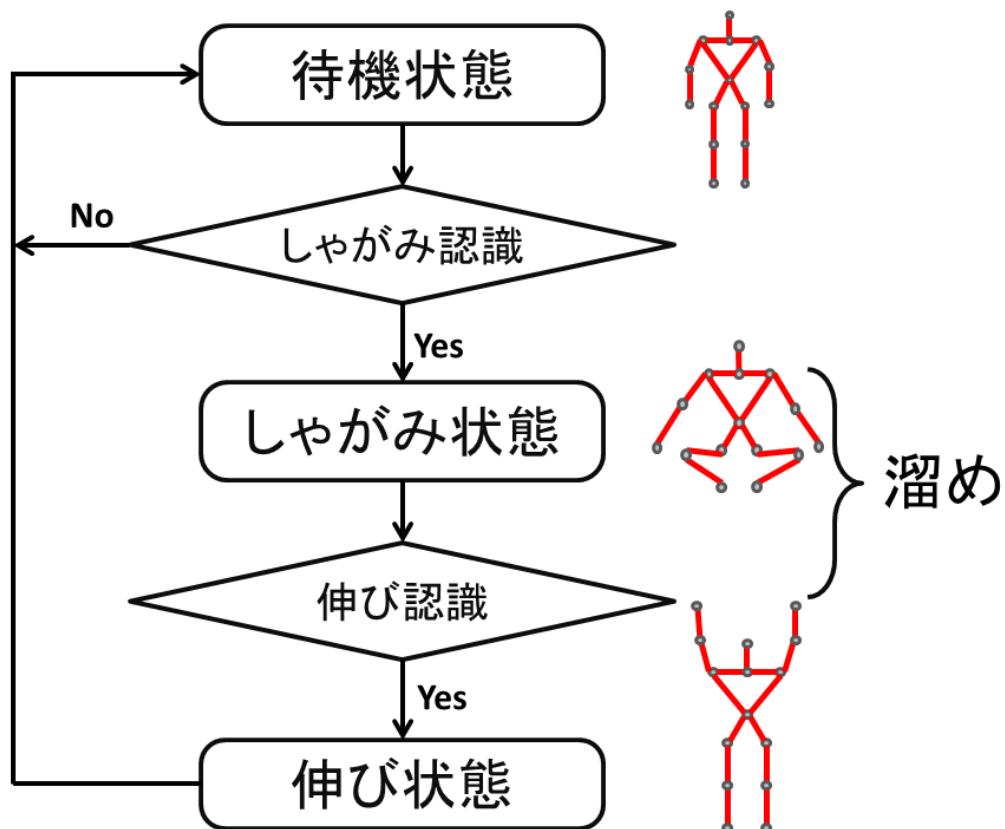


図 4.3 CartooNect の状態遷移図

4.1.4 まとめ

一連の動作から、溜め動作という静止状態を認識が出来るようになった。国際・国内展示やオープンキャンパスで展示をおこない、300 人以上のプレイヤに体験していただけた。

4.2 没入感の高いリアルタイムアニメーションツール「GAMIC」

4.2.1 GAMIC のコンセプト

GAMIC は「Game Action Motion Interaction Controller」の略で、プレイヤの動作とキャラクタアニメーションの再生タイミングを調整、誇張されたキャラクタアニメーションの再現を、プログラミングなしに容易に開発することのできるアルゴリズムおよびツールである。

骨格情報を用いたゲームではユーザの動作をすばやく認識し、対応したキャラクタアニメーションを行う必要がある。ユーザが動きを終えた時点で動作を認識した場合、対応したキャラクタアニメーションが再生されるまでに遅延が発生し、ユーザはキャラクタとの一体感、ゲーム中への没入感を感じることが出来ないだろう。認識の精度の調整が必要となるが、低すぎる場合は誤認識が多くなり、精度が高すぎる場合には、ユーザによる動作の再現が難しくなる。また姿勢認識の精度の調整はプログラマやデザイナ、モーションインターラクションデザイナなど多くの人が必要となってしまう。

姿勢認識を単純にするため、評価関数によって認識を行う。キャラクタアニメーションと関連付けられたターゲット姿勢の骨格情報と、ユーザの骨格情報を比較し類似度を求める。ターゲット姿勢の骨格情報ごとに設定した閾値を、評価関数によって求めた類似度が満たしていた場合、「ターゲット姿勢をした」と認識する。各ターゲット姿勢の骨格情報の閾値を操作することで、精度を変更できる。ターゲット姿勢の骨格情報や閾値は外部ファイルに出力でき、プログラマが個々に値を設定する手間を省くことが出来る。

4.2.2 姿勢認識アルゴリズム

姿勢認識アルゴリズムは、プレイヤの骨格情報と、設計者が用意したターゲットの骨格情報を比較する。

プレイヤとターゲットの各関節から内積をとり、内積の総和を求める。内積の総和評価関数(式 4.1)によって求める。現在のプレイヤの骨格情報を V 、ターゲット姿勢の骨格情報を T とするとき、評価関数 f は(T, V)になり-1.0 ~ 1.0 を出力する。

$$f(T, V) = \frac{1}{k} \sum_{i=1}^k \left(\frac{T_i \cdot V_i}{\|T_i\| \cdot \|V_i\|} \right) \quad (k = \text{関節数}) \quad (4.1)$$

ターゲット姿勢の骨格情報ごとに用意された閾値 P と評価関数 f によって求めた値と比較し、認識の判定(式 4.2)を行う。評価関数 f が閾値 P を満たすことで、閾値 P に関連づかされているターゲット姿勢を行ったと認識される。閾値 P を変更することで、認識精度を制御することができ、閾値 P を 1.0 に近づけるほどユーザの骨格情報をターゲット姿勢の骨格情報に極めて近づけなければ判定されない。逆に閾値 P を小さくしていくと、誤認識が多くなる。

$$f(T, V) > P \quad (4.2)$$

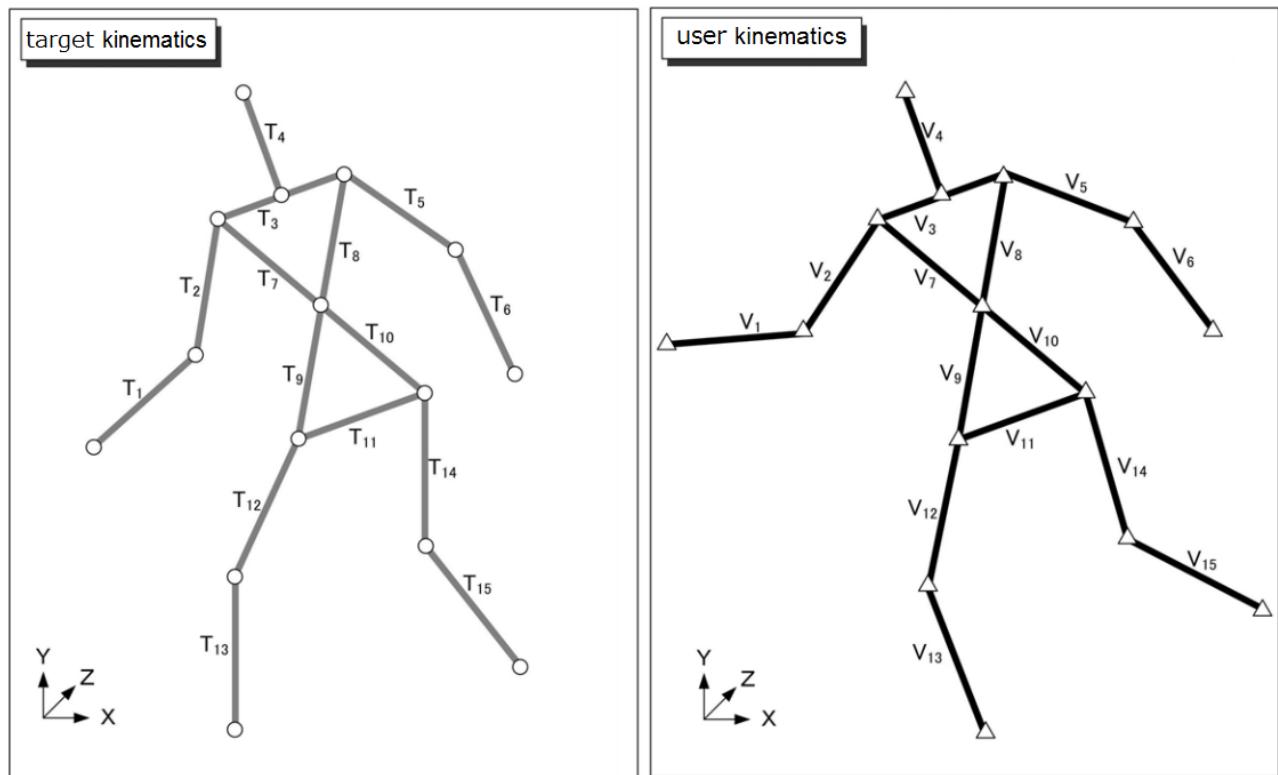


図 4.4 ターゲット姿勢 T_i (左) とユーザ V_i (右) の骨格情報

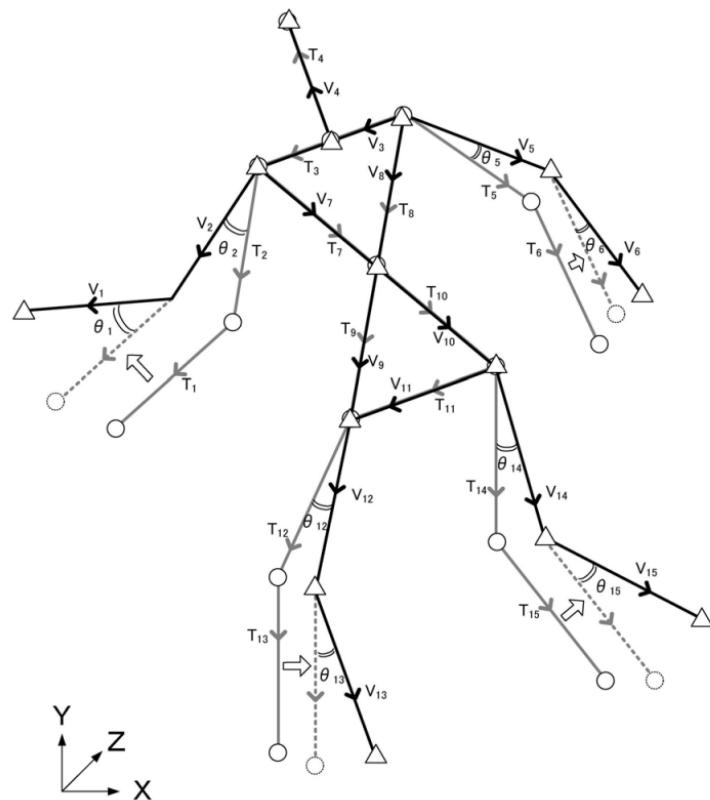


図 4.5 キネマティクスの比較

4.2.3 ツール「GAMIC」

プレイヤの骨格情報と，事前に外部ファイルに用意しておいた数種類のターゲット姿勢の骨格情報との一致率を，評価関数（式 4.1）によって評価値-1.0 ~ 1.0 で求める。評価値と，姿勢ごとに設定された閾値を比較し，評価値が閾値を満たしていた場合，その姿勢をしていると認識するアルゴリズムを開発した。

GAMIC の操作には Wii リモコンを用いており，ターゲット姿勢の骨格情報の撮影，閾値の調整が行える。Wii リモコンによって閾値を調整する事で，デザイナー一人でも認識精度を調整することが出来る。調整した閾値は外部ファイルに出力できるので，プログラミングを行う必要がない。

図 4.6 が GAMIC の画面である。画面中の $P1$ と $P2$ は，設定した閾値で現在は一致率が 50% を超えた場合，姿勢を認識する。Wii リモコンの上下を押すことで $P1$ ，左右を押すことで $P2$ の調整ができる。

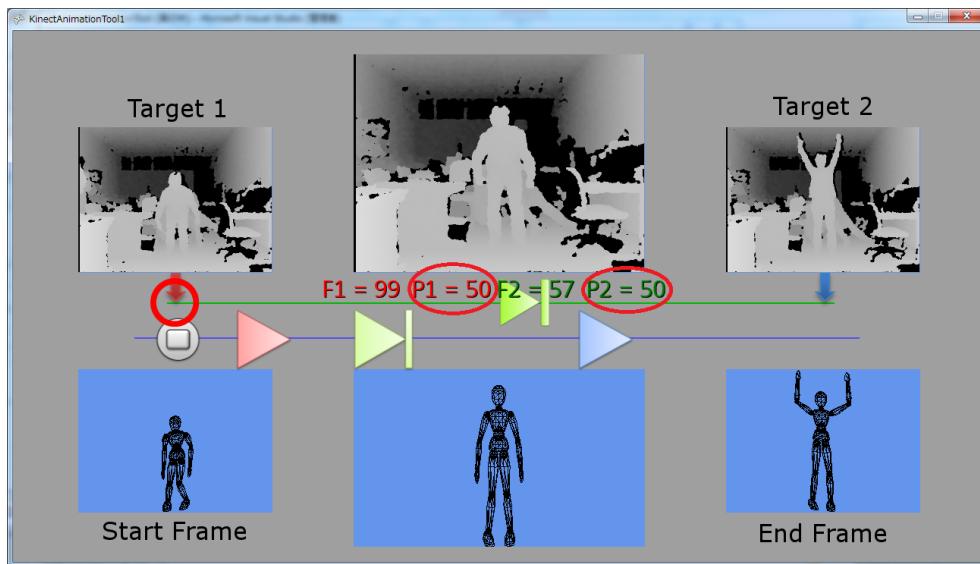


図 4.6 GAMIC のスクリーンショット

4.2.4 まとめ

姿勢認識をアルゴリズム化を行ったことで，1つの評価関数によって複数の姿勢を認識できるようになった。また閾値の操作によって，認識精度を調整をプログラミングを行うことなくできるようになった。

4.3 AccuMotion

4.3.1 AccuMotion 概要

AccuMotion は，CartooNect の溜め認識，GAMIC の姿勢認識アルゴリズムを組み合わせた操作ツールである。私が CartooNect や GAMIC といった，ゲームコンテンツやツールを開発

してきて、Kinect をゲームコンソールのインターフェースだけでなく、応用して様々なことに使えるのではないかと感じた。そこでパワーポイントなどのアプリケーション操作ツールを開発しようと考えた。AccuMotion では、キーボード入力の上下左右にあたる入力を再現する。

4.3.2 AccuMotion のアルゴリズム

AccuMotion は 6 つの状態を溜め認識、姿勢認識によって遷移させている。各状態の対応姿勢は図 4.7 にまとめた。

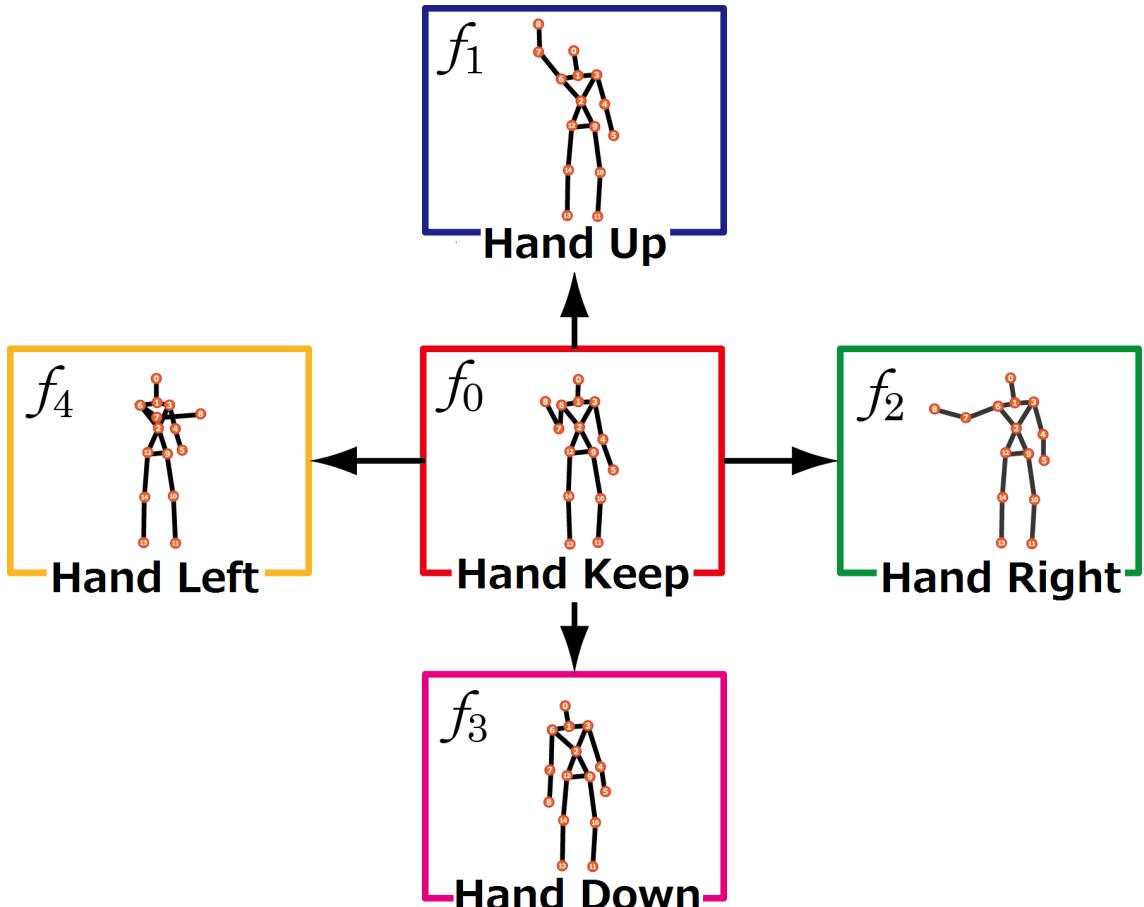


図 4.7 AccuMotion の入力姿勢

待機状態から f_0 の姿勢をとることで、他の $f_1 \sim f_4$ の認識ができるが、 f_0 は溜め動作を行う必要がある。溜め動作によって、自然な動作と操作動作を区別することが出来る。 f_0 が認識された場合、状態が Hand Keep となり上下左右へ入力することができる。AccuMotion の状態遷移図を図 4.8 に示す。

4.3.3 実験

この実験は開発したアルゴリズムで、いかなるユーザでも正しい入力操作を行うことができるか検証する。実験環境は図 4.9 のように設定した。Kinect を 75cm の台に設置し、被験者は Kinect から全身がキャプチャーされるように 2.5 ~ 3.0m ほど離れて入力動作を行う。

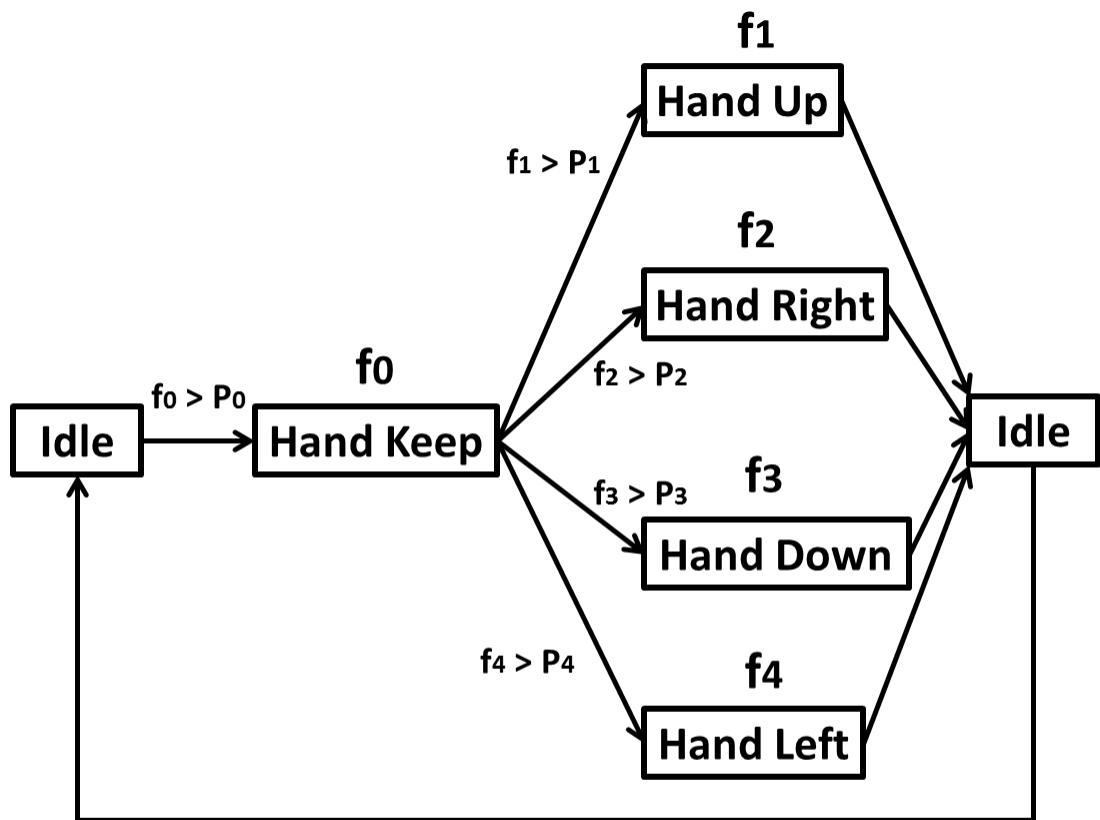


図 4.8 AccuMotion の状態遷移図

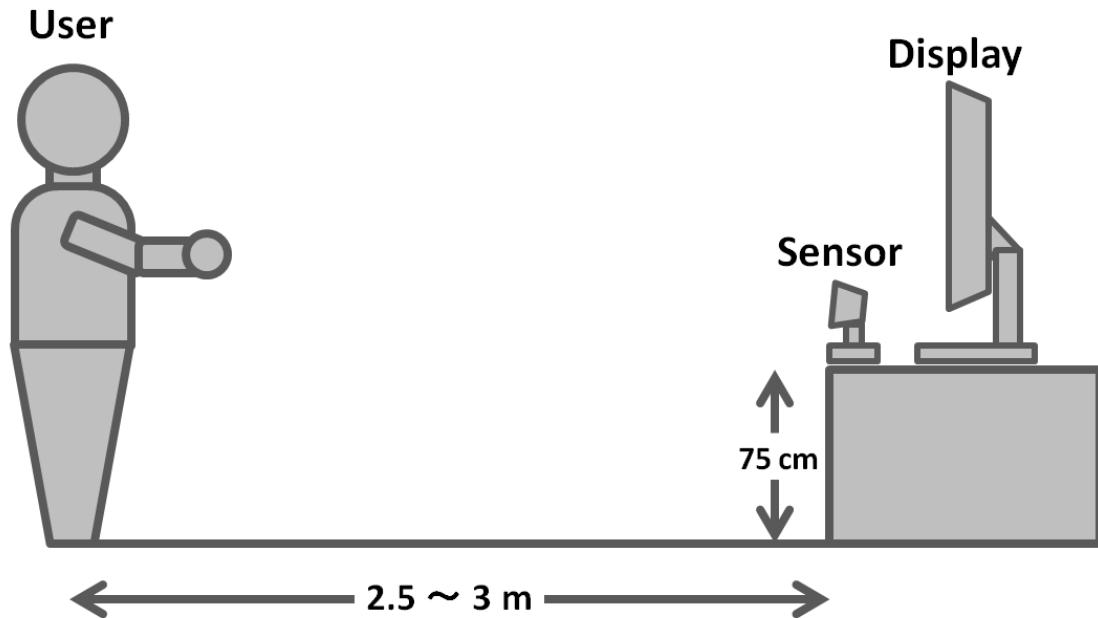


図 4.9 AccuMotion 実験図

被験者は女性 1 人 , 男性 12 人 , 年齢 20 ~ 38 才を対象とした . 事前に被験者には , 1 度だけ実験プログラムの入力操作方法を説明する . この段階で , それぞれの入力動作を 1 度体験してもらった . 実験プログラムの入力方法は , まず ”溜め” の姿勢 (f0 の動作) を行う . この動作を 1 秒間行うと , 「溜め」 (ユーザーが意図して入力動作を行った) の姿勢と認識し , 上下左右の方向へ入力が可能になる . ディスプレイに表示した 4 つの矢印の方向に , 出題通りに左から順番に入力していく (図 4.10) . 4 方向への入力を 1 問と設定し , 連続して 5 問行い各問題にかかった入力時間を計測するし , 終了後に口頭で印象を収集する . 実験結果は次章の 「 AccuMotion の実験結果 」 の節にて記述する .

姿勢判断に用いる入力姿勢データは , 被験者の体格や性別によって変更は行わず , すべての被験者が同じデータを用いる . 本実験を行った時の各入力姿勢評価関数の閾値を表 4.2 に示す .

表 4.2 実験に設定した閾値

評価関数	閾値
f0 (HandKeep)	94
f1 (HandUp)	94
f2 (HandRight)	95
f3 (HandDown)	94
f4 (HandLeft)	95

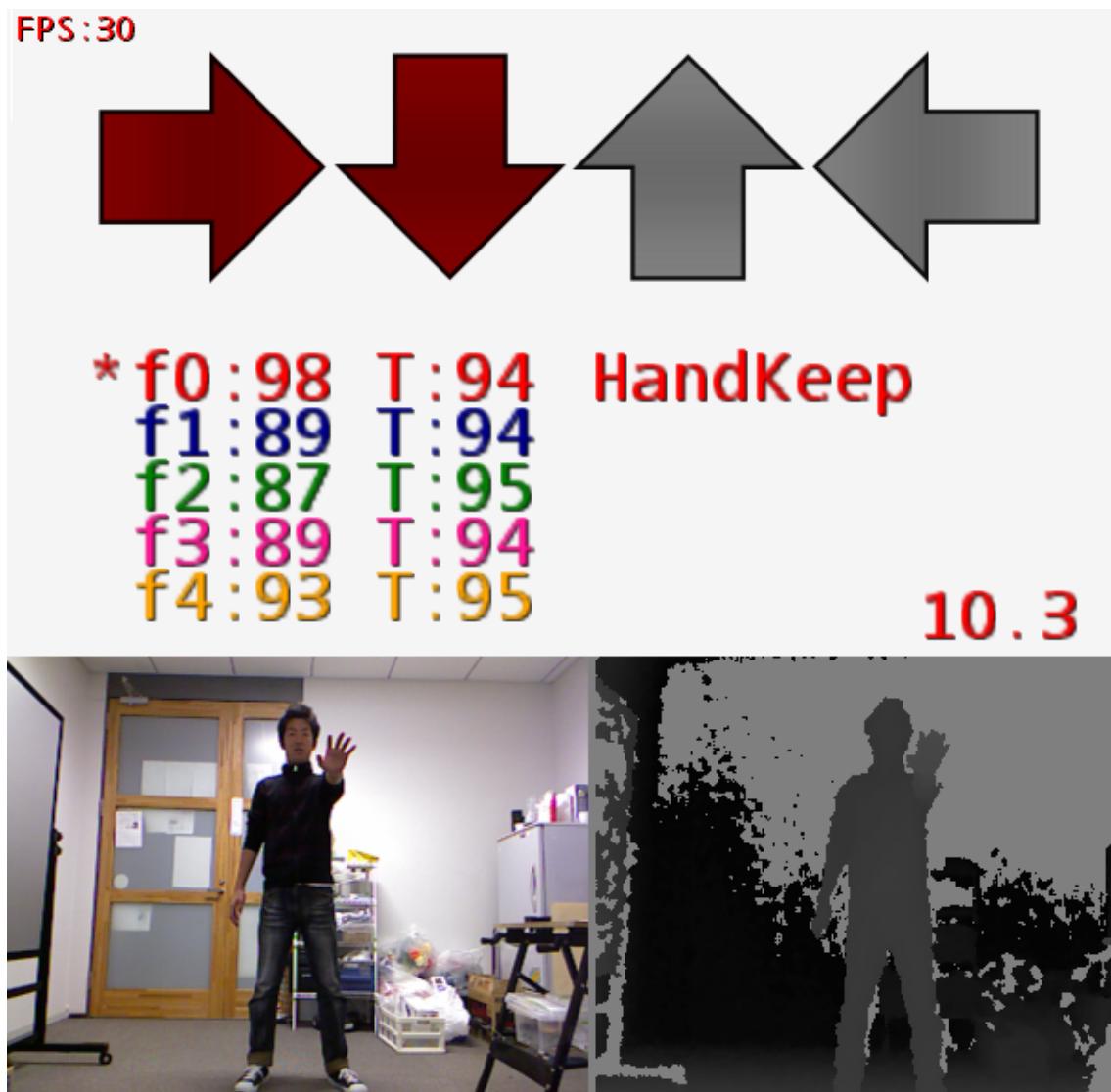


図 4.10 AccuMotion のスクリーンショット

5. 評価

5.1 CartooNect の評価

2011年3月フランスのラバル市で行われた，VRIC2011 (Virtual Reality International Conference)において展示を行った。展示を5日間行ったところ，メッセージ画を160枚以上集めることが出来た。文化の違いによるメッセージのとらえ方や，言語に関係なく，製作意図である支援喚起をVR技術を活用して実現することができた。

日本での展示は，2011年5月21日に横浜で開催された「科学のひろば」で展示をした。コンテンツを体験しながらヒマワリの効果を知ると「家でヒマワリを育ててみたい」(小学5年生)といった意見をもらうことが出来た。また「狭い空間でも激しい運動が出来てよい」(保護者)といった意見ももらうことができ，VR技術を使ったシリアルスゲームの可能性を見つけることができたと考えている。



図 5.1 CartooNect の体験の様子

5.2 GAMIC の評価

2011年8月7~11日にカナダ、バンクーバーで開催されたSIGGRAPH 2011のポスターセッションで発表を行った。Kinectを使ったコンテンツを開発する上で，多くの開発者が調整に苦労する箇所をプログラミングすることなく，評価関数と閾値によって制御することが評価

を得たようであった。展示を行って Xbox Live 開発関係者に興味を持ってもらい意見を聞くことができた。

両膝の角度が一定以上の場合、しゃがみを認識するとしていた CartooNect に GAMIC のアルゴリズムを組み込んだところ、認識精度の向上が見られた。また、腕による十字移動が可能になった（図 5.2）。



図 5.2 GAMIC のアルゴリズムを組み込んだ CartooNect

5.3 AccuMotion の実験結果

実験の結果、4 方向への入力を 1 問とし、T1 から T5 まで行った結果、T1 の平均入力時間は 14.2 秒であったが、T5 の平均入力時間は 10.9 秒と入力時間が短くなっている（図 5.3）。この現象は被験者の約 92% が当てはまる。多くの被験者は平均 7.5 秒で答えることができ、標準偏差は 11.6 秒であった。多くの被験者が入力方法を理解し、速く、正確に行うことが出来たといえる。

実験終了後口頭にて意見を得たところ、左右 (f2, f4) の認識精度が対象ではなくやりづらかった、溜めのポーズ (f0) が分からなかったといった意見が出た。

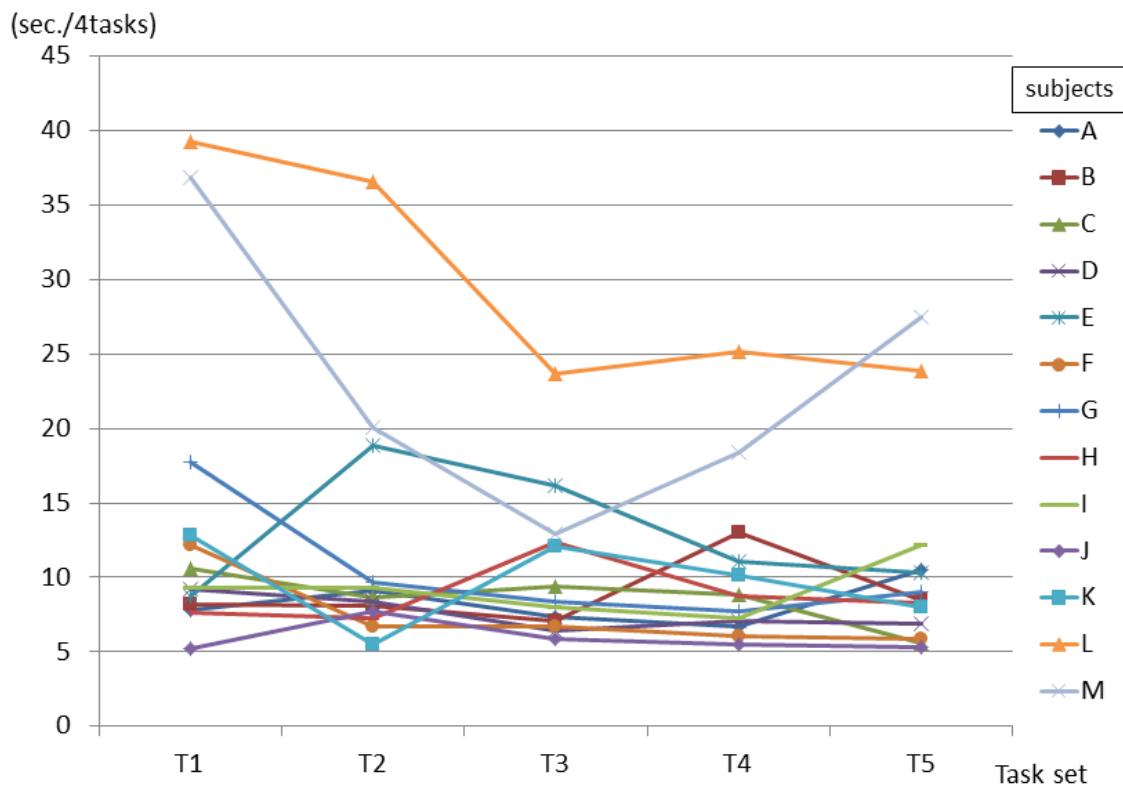


図 5.3 問題回答にかかった時間

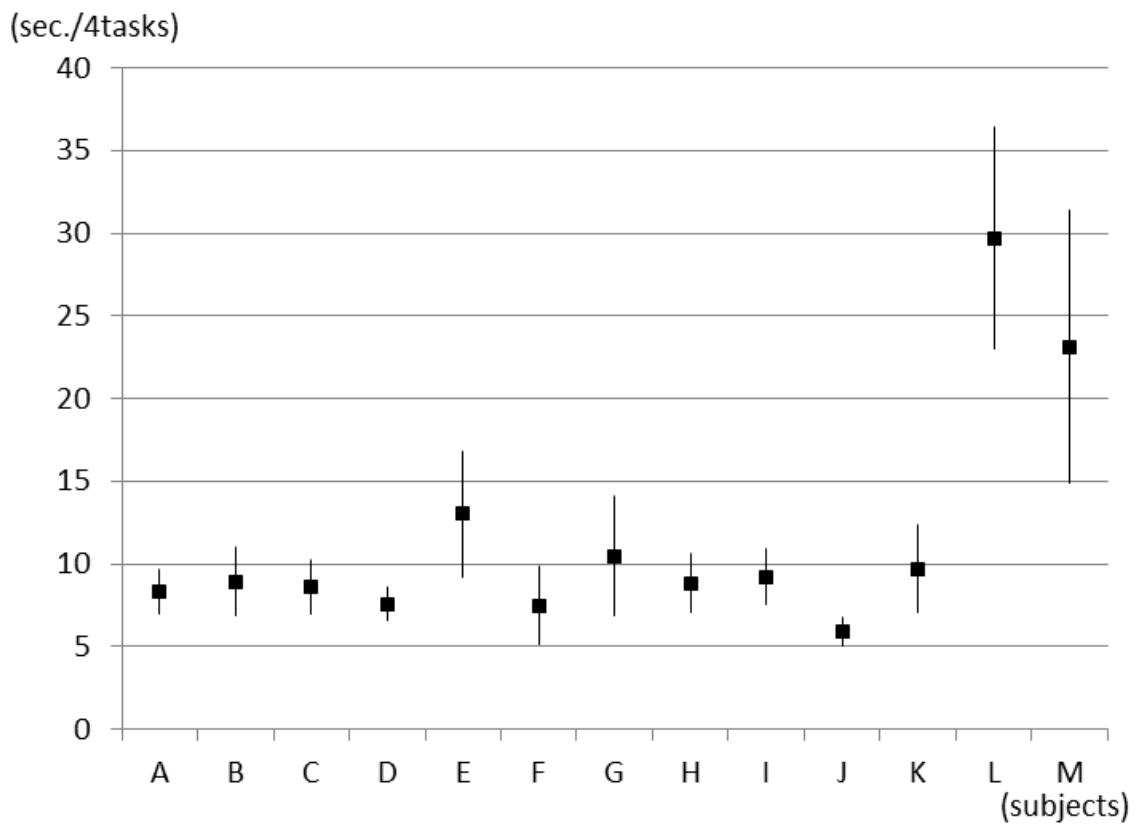


図 5.4 一問回答するのにかかった平均時間

6. 展望

今後の展望としては、デジタルサイネージや家電製品の操作を行うことが出来るのではと考えている（図6.1）。また、物に直接触れる必要がないので医療現場、食品関係の現場など衛生を重視する環境での使用が考えられる。

技術的な課題としては、なにも装着物がないので本当に認識しているのかわからないといった部分である。こうした場合音声や、画面に情報を表示して使用者に分かりやすくなればないと考える。



図6.1 NUIによるテレビ操作のイメージ

謝辞

本論文の執筆にあたり、日々ご指導頂いた指導教員の白井暁彦准教授に深く感謝します。本研究を進めるに際して、様々なご指摘をくださいました小坂崇之助教授、GAMIC を共同で開発した服部元史研究室の三角甫君、実験および本研究にご協力いただいた各位にこの場を借りて感謝の意を表します。

参考文献

- 1) 増井 俊之 : Interface2012年1月号 , pp.30-39 , CQ出版社 (2012)
- 2) 加藤博一 , Mark Billinghurst , Ivan Poupyrev , 鉄谷信二 , 橋啓八郎 : 拡張現実感技術を用いたタンジブルインターフェース , 芸術科学会論文誌 , Vol.1 , No.2 , pp.97-104 (2002)
- 3) 白井暁彦 , 小坂崇之 , 藤村航 , Interface2012年1月号 , pp.85-86 , CQ出版社 (2012)
- 4) 西村考 , 小野憲史 : キネクトハッカーズマニュアル , pp.12-18 , 株式会社ラトルズ (2011)
- 5) Tech Fielders セミナー東京 「Kinect for Windows SDK をつかおう！」 ,
<http://www.microsoft.com/japan/powerpro/TF/seminar.mspx#s125>
- 6) Kinect を使ったアプリケーション開発概要 ,
<http://www.dceexpo.jp/program/others/detail.php?lang=jp&code=OT201117&category=9>
- 7) Kinect for Windows SDK , <http://www.microsoft.com/en-us/kinectforwindows>
- 8) Kinect for Windows SDK Blog , <http://blogs.msdn.com/b/kinectforwindows/archive/2012/01/09/kinect-for-windows-commercial-program-announced.aspx>
- 9) OpenNI , <http://www.openni.org>
- 10) OpenNI Documentation , <http://openni.org/Documentation>
- 11) 中村薰 : KINECT センサープログラミング , p.10 , 秀和システム (2011)
- 12) BLEIWEISS A. et. al.: Enhanced interactive gaming by blending full-body tracking and gesture animation , ACM SIGGRAPH ASIA 2010 Sketches , Article No.34 , ACM (2010)
- 13) Kinect でなりきりウルトラセブン ! , <http://www.youtube.com/watch?v=eCbURRDUUdI>
- 14) Kinect でかめはめ波!改 コード公開版 , <http://www.youtube.com/watch?v=r0l18SWKd88&feature=related>
- 15) Team Lab Mirror , <http://dev.team-lab.com/index.php?itemid=252>
- 16) 柴田智広 , 船谷浩之 , 中村彰宏 , 大林千尋 : Interface2012年1月号 , pp.107-111 , CQ出版社 (2012)
- 17) "NAIST Ballpool" for NAIST , <http://www.youtube.com/watch?v=LuNdc-Ejyjg>
- 18) デジタルゲームの教科書制作委員会 : デジタルゲームの教科書 , pp.229-246 , ソフトバンククリエイティブ株式会社 (2010)
- 19) 白井暁彦 : エンタテイメントシステム , 芸術科学会論文誌 , Vol.3 , No.1 , pp.22-34 (2004)
- 20) 農林水産技術会議 , <http://www.saffrc.go.jp/docs/press/110914.htm>
- 21) FUJIMURA Wataru , IWADATE Shoto , SHIRAI Akihiko : CartooNect:Sensory motor playing system using cartoon actions , Virtual Reality International Conference 2011 (VRIC2011) , 6-8 April 2011 , Laval , France

- 22) FUJIMURA Wataru , MISUMI Hajime , KOSAKA Takayuki , HATTORI Motofumi , SHIRAI Akihiko : Development of serious game which use full body interaction and accumulated motion , NICOGRAPIH International 2011 , 10-11 June 2011 , Kanagawa , Japan
- 23) Hajime Misumi , Wataru Fujimura , Takayuki Kosaka , Motofumi Hattori , Akihiko Shirai : GAMIC:Exaggerated Real-Time Character Animation Control Method for Full-Body Gesture Interaction System , ACM SIGGRAPH 2011 , 7-11 August 2011 , Vancouver , Canada
- 24) 藤村航 , 三角甫 , 小坂崇之 , 白井暁彦 : 全身運動を中心とした震災復興を伝えるシリアルゲームの開発 , 第16回日本バーチャルリアティ学会大会誌論文集 , 2011年9月 (2011)