

修 士 論 文

仮想都市環境における人間と仮想空間の
インタラクションインタフェースの構築

北陸先端科学技術大学院大学
情報科学研究科情報システム学専攻

酒井潤

2004年3月

修士論文

仮想都市環境における人間と仮想空間の インタラクションインタフェースの構築

指導教官 堀口 進 教授

審査委員主査 堀口 進 教授

審査委員 Shen Hong 教授

審査委員 松澤 照男 教授

北陸先端科学技術大学院大学
情報科学研究科情報システム学専攻

210039 酒井潤

提出年月: 2004 年 2 月

概要

近年、コンピュータは格段に高性能になり、仮想現実（バーチャルリアリティ）の技術が進歩してきている。バーチャルリアリティ技術とは、コンピュータ内に電子的な物体等を構築することにより、人間が実世界で体験するのと同様な体験を可能にする技術である。例えば、都市や住宅を設計する場合、等身大の仮想都市を構築して、あらかじめ安全性や快適性の評価を行うことができる。この都市環境の仮想体験をより現実感のあるものとするためには、実際の都市と同じ等身大での体験ができることが望ましい。仮想環境に没入できるような視覚情報を提示するだけでなく、ユーザとコンピュータ間のインタラクションインタフェースも違和感のないものにする必要がある。

従来の仮想3次元空間内を移動するためのインタフェースとしてはさまざまなものが提案されている。より移動感覚を自然に提示するためのものとして手の振りをインタフェースとする研究が行われている。しかし、移動のために手を用いると、それ以外の他の目的に手を使うことができないという欠点がある。また、体重移動による仮想空間内移動インタフェースを提案した研究も行われているが、人間が自然に行っている動作ではないので直感性に欠ける。特に都市設計のための仮想都市環境システムでは、評価の都市内を移動する必要があり、安全性や快適性に問題があれば、設計物や置物の配置を変える必要がある。このような仮想都市環境システムでは、少なくとも仮想環境内を移動し、仮想物体をつかみ別の場所へ移動することが必要となる。しかし、従来研究のインタフェースでは、不十分である。

本研究では、人間が実世界で行う動作と同様な動作によりインタラクションが可能なインタフェースを設計構築する。具体的には、人間の歩行動作により仮想都市内を自由に移動し、人間の手により仮想物体を操作するインタフェースを提案する。そして、モーションキャプチャを用いて歩行ならびに仮想物体操作インタフェースを実装し、評価実験によりこれらの仮想都市環境におけるインタラクションインタフェースの有効性について詳しく検証する。さらに、仮想都市空間に自分や他人の分身であるアバタを導入し、アバタとのコラボレーションを実装してその有効性についても議論する。

目次

第1章	序論	1
1.1	本研究の背景と目的	1
1.2	本論文の構成	2
第2章	仮想現実技術と仮想都市環境	3
2.1	はじめに	3
2.2	仮想現実技術	3
2.3	仮想都市環境	4
2.4	インタラクションインタフェース	4
2.5	まとめ	7
第3章	仮想都市環境における移動インタフェース	8
3.1	はじめに	8
3.2	実際の歩行動作の計測	9
3.2.1	計測方法	9
3.2.2	歩行動作の分析	11
3.3	歩行速度の推定	13
3.4	一步目の予測	14
3.5	拡張歩行	15
3.6	仮想都市映像の拡大縮小による拡張歩行	16
3.7	身体方向を利用した仮想空間内移動方向制御	16
3.8	まとめ	18
第4章	仮想都市環境における仮想物体操作インタフェース	22
4.1	はじめに	22
4.2	仮想物体操作の過程	22
4.3	まとめ	24
第5章	仮想都市環境インタラクションシステム構成	26
5.1	はじめに	26
5.2	入出力装置	26

5.2.1	Flock of birds	26
5.2.2	CAVE	27
5.2.3	シャッターメガネ	29
5.2.4	立体視対応・超高精細 CRT プロジェクタ	29
5.2.5	Onyx3200	30
5.3	システム構成	31
5.3.1	Motion Star を使用したシステム構成	32
5.3.2	Flock of Birds を使用したシステム構成	33
5.3.3	CAVE への仮想都市の表示	34
5.3.4	仮想都市の表示のためのシステム内のプロセス	34
5.3.5	背後の仮想空間へ移動する場合	34
5.3.6	CAVE のスクリーンへの接触問題	35
5.4	移動インタフェースの速度制御と方向制御の評価実験	35
5.4.1	足踏み歩行による目的地までの速度制御	35
5.4.2	ターゲット追跡のための足踏み歩行による速度制御	36
5.4.3	足踏み歩行による目的地までの方向制御	36
5.4.4	足踏み歩行による経路上の方向制御	36
5.5	3DWAND との比較	37
5.5.1	3DWAND との比較実験	37
5.5.2	3DWAND の操作方法	37
5.5.3	3DWAND との比較実験結果	38
5.6	まとめ	39
第 6 章	仮想都市空間におけるアバタ構築とコレボレーション	54
6.1	はじめに	54
6.2	アバタのある仮想都市空間システム構成	55
6.3	アバタ作成のための人体構造の定義	55
6.4	仮想空間における人体モーションの合成	56
6.5	仮想都市空間でのアバタとのコラボレーション	58
6.6	考察	58
6.7	まとめ	59
第 7 章	まとめ	64
7.1	まとめ	64

第1章 序論

1.1 本研究の背景と目的

近年、コンピュータは格段に高性能になり、仮想現実（バーチャルリアリティ）の技術が進歩してきている。仮想現実とは、人間の外界認識を司る視覚、聴覚、触覚等の器官に対して、コンピュータによる合成情報を提示し、それによって人間周囲に仮想的な世界を作り上げるための技術である。このバーチャルリアリティの技術を用いることにより、人間が実世界で体験するのと同じように仮想の世界で体験することが可能になる。例えば、都市や住宅などを設計する場合、等身大の仮想デジタルシティを構築することで、あらかじめ安全性や快適性の評価を行うことができる。このような仮想環境では、実世界で人間が見たり動作したりすることと同じような体験ができることが要求されており、このような仮想環境では、実世界の人間の動作と同じようにインタラクションすることが要求されており、少なくとも仮想環境内を移動し、仮想物体をつかみ別の場所へ移動するということが絶対不可欠になる。

従来の仮想3次元空間内を移動するためのインタフェースとしてはさまざまなものが提案されている。久木元[1]は、より移動感覚を自然に提示するために手の振りをインタフェースとした手法を提案した。しかし、移動のために手を用いると、それ以外の他の目的に手を使うことができないという欠点がある。また、志和[2]らは、体重移動による仮想空間内移動インタフェースを提案したが、人間の自然の動作またはそれに近い動作でないで直感性に欠ける。特に仮想都市環境において、自然な動作でインタラクションするインタフェースとしては、不十分である。

仮想環境と自然にインタラクションするには、実世界の人間の動作を仮想世界に反映させる必要がある。そこで本研究では、人体各部位に特徴点を設置し、その各点の3次元座標を使って仮想都市内を自由に移動し、仮想物体を操作する手法を提案する。

そのために、モーションキャプチャを用いることによって、人間動作の3次元データを正確に取得し、人体の動きに応じて仮想空間と直感的にインタラクションする新しいインタフェースについて述べる。実際に仮想空間内を自由に移動するには、速度制御法と方向制御法が必要になる。そこで、モーションキャプチャによって得られる3次元位置データから、移動の際の速度制御および方向制御を提案する。仮想都市空間内の移動インタフェースには、人間の歩行動作に近い足踏み歩行を用いる。その場での足踏み動作をインタフェースに用いる利点としては、人間の移動手段としての歩行動作に近く、実際に移動するための仮想環境と同じ広さの空間が必要でないことである。さらに、仮想都市環境で

あるという特徴を生かして、高速に移動する場合、歩幅を大きくすることで歩行速度を速くする機能も付加した。他の都市へ移動する長距離移動の場合は、仮想都市の縮尺を変えることで容易に移動する方式も導入する。本研究では、移動操作を足に担わせることにより手で他の操作が可能になるため、作業の基礎動作となるオブジェクトをつかみ移動する機能も実装する。仮想物体の操作法も、人間が実世界で行う動作に近い方が望ましい。本研究では、握るという動作により物体を選択する仮想物体移動法を提案する。ユーザの手首と手先にそれぞれ1つ座標を設置したモデルを考え、ベクトルの大きさを計算する。そのベクトルの大きさがある閾値を越えた場合に握ったと判定する。仮想物体と手の接触判定は、仮想オブジェクトを直方体の領域で取り囲み、この直方体の中に、腕先に設置した座標が入ることで接触していると判定する手法を採用した。そして、提案手法を用いた仮想都市インタラクションシステムを構築し、仮想都市内へ自分や他の分身であるアバタを導入することで、仮想都市内のアバタとコラボレーションを実現する。

1.2 本論文の構成

以降、第2章では、仮想都市で実現するためのインタラクション機能を明らかにする。また、従来研究のインタフェースを説明し、仮想都市内を自然な動作でインタラクションをするのに不十分であることを明らかにする。第3章では、実際の歩行動作の計測結果から、足踏み動作による仮想空間内の歩行速度制御と移動方向制御の手法について議論する。さらに、仮想都市環境であるという特徴を生かした高速な空間移動方式についても概説する。第4章では、人が実世界で行う動作のように、仮想物体を操作する手法を提案する。第5章では、提案した手法を用いた仮想都市インタラクションシステムについて述べる。次に、実現した仮想都市環境システムの中で、従来研究であげた3DWANDと提案したインタフェースとの比較実験を行い、本インタフェースの有効性を示す。第6章では、都市生活者の視点から都市を評価するための仕組みを提案する観点から、仮想都市空間へ自分や他の分身であるアバタの導入方法について述べる。第7章では、本論文をまとめ、結論とする。

第2章 仮想現実技術と仮想都市環境

2.1 はじめに

都市環境の仮想体験をより現実感のあるものとするためには、実際の都市と同じ等身大での体験ができることが望ましい。臨場感のある等身大の仮想都市環境により、その都市を仮想体験することで、都市の問題点を体感することができ、都市の評価、設計に役立つと期待されている。

このような仮想都市空間では、なるべく実空間と同じ条件下で行うのが望ましいが視覚情報のみで頼るリアリティには限界がある。高い没入感を提示するという事は実空間で感じる感覚と違和感を少なくする必要がある。そのためには提示情報がよりリアリティを増しているように、ユーザとコンピュータへのインタラクションインタフェースも違和感のないインタラクションインタフェースにする必要がある。実世界と差異のないインタラクションインタフェースとは人間が実世界で行う動作と同じような人の身ぶりであると考えられる。

本章では、仮想都市で実現するためのインタラクション機能を明らかにする。また、従来研究のインタフェースを説明し、仮想都市内を自然な動作でインタラクションをするのに不十分であることを明らかにする。続く2.1では、仮想現実技術について説明する。2.2では、仮想都市環境の重要性を述べる。2.3では、仮想世界とのインタラクションのための従来のインタフェースについて概説する。2.3では、まとめを述べる。

2.2 仮想現実技術

近年コンピュータは格段に高性能になり仮想現実技術においてリアリティを高めるために高品質のグラフィックスを使用している。グラフィックスの精度が向上すればユーザの仮想空間への没入感はさらに高くなる。しかし、視覚的に臨場感が増せばVR酔い(Simulator Sickness)など望ましくない影響もユーザに与えることになる。このような環境は視覚情報と体性感覚の整合性がとれていないためと考えられる[3]。特に仮想空間内でシュミレーションやトレーニングといった作業を行う際に、なるべく実空間と同じ条件下で行うのが望ましいが視覚情報のみで頼るリアリティには限界がある。高い没入感を提示するという事は実空間で感じる感覚と違和感を少なくする必要がある。そのためには提示情報がよりリアリティを増しているように、ユーザとコンピュータへのインタラクションインタフェースも違和感のないインタラクションインタフェースにする必要がある。実世界と差

異のないインタラクションインタフェースとは人間が実世界で行う動作と同じような人の身ぶりであると考える。

2.3 仮想都市環境

都市設計において重要な点は、その都市に住む人間のために安全に快適な環境を実現することである。しかし、都市の完成後に問題点が発生した場合、都市を作り変えることは、困難なことである。そのため、設計時にあらかじめ安全性や快適性の評価を行う必要がある。

従来、このような評価は、異なる専門分野を持つ複数の専門家達の間で検討されてきた。しかし、その都市で実際に生活する一般市民と専門家との間での認識の違いから、必ずしも正確な評価とならない場合もあり、後から予想外の問題点が発生することもある。そのために、設計時には専門家の意見だけでなく、一般市民の意見も取り入れることが望まれる。このとき、設計における専門的な事柄が、一般市民に理解されにくいこともあるため、専門知識を必要としない、誰でも簡単に評価に参加することができるような方法が必要となる。

この方法の一つとして、仮想現実技術（バーチャルリアリティ）の技術を利用する方法がある。都市環境の仮想体験をより現実感のあるものとするためには、実際の都市と同じ等身大での体験ができることが望ましい。臨場感のある等身大の仮想都市環境により、その都市を仮想体験することで、都市の問題点を体感することができ、都市の評価、設計に役立つと期待されている。

2.4 インタラクションインタフェース

仮想都市環境を体験するには、仮想空間を自由に移動し、仮想物体を操作する必要がある。ここでは、従来のインタフェースの機能、特徴を整理する。

仮想3次元空間内とインタラクションインタフェースとしては様々なものが提案、実現されている。マウスやジョイスティックのように手で操作するインタフェースが圧倒的に広く普及している。身体の他の部位を使って操作するインタフェースの研究例もいくつか報告されている。また、体感ゲーム機分野ではそれぞれのゲームに合わせていろいろなインタフェースが開発されている。

- マウス

マウスは最も標準的なインタフェースデバイスである。システムによる差があるものの、マウスカーソルで画面上部を押すと前進、下部を押すと後退、右または左を押すと押した方向に進行方向を変更するというようなインタフェースが一般的である。画面の上下に方向を割り当てるのではなく、ボタンを押し下げて上方に移動すると前進というように

ボタンを押し下げた位置からの変位によってコントロールする方法も一般的である。このようなマウス単独によるインタフェースのほかに、グラフィカルユーザインタフェースとの組み合わせによって移動を指示するインタフェースも一般的である。例えば、画面上に表示されたハンドルを操作することで移動を指示するインタフェースなどがある。

- ジョイスティック

ジョイスティックでは、前に倒すと前進し横に倒すと倒した方向に転回するというように、行きたい方向にスティックを倒すことによって移動する方向をコントロールする。力をかけなければ、スティックはもとの位置に復帰し、移動は停止する。行きたい方向にスティックを倒すという操作は直感的で、理解しやすいインタフェースと言える。3次元マウスと呼ばれるデバイスもこれに似ている。軸で固定された球状のつまみを前後、左右、上下に押すことで対応する方向に移動することができる。またつまみをねじることで向きを変えることができる。かけた力を解くと、もとの位置、向きに復帰し移動は止まる。

- 3D-Wand

3D-Wandは手に持つことができる小型の棒状のデバイスで、位置と方向を検出するセンサと数個のボタンを備えている。3次元空間中の位置や方向を指しながらボタンによる指示ができるので、例えば仮想空間中の物体をつかんで移動したりすることができる [4]。このデバイスを用いて仮想空間内を移動するインタフェースでは、例えば、右に進むには棒を右に向けてボタンを押す、右に転回するには右ボタンを押すというような操作を行う。また、ハンドジェスチャを入力、認識し、歩いたり走ったりという指示を出すインタフェースも開発されている [6]。

手を用いるインタフェースは理解しやすく、また疲労が少ないという点で、扱いやすいインタフェースであるといえる。しかし、仮想環境内では移動だけではなく、いろいろな操作を行う場面も考えられる。その操作の多くは手を用いることが想定される。したがって、手を移動のために用いるのは好ましくない。また、仮想環境の中で自分が移動しているように感じる没入感があまり得られない。

- 歩くインタフェース

我々は人間は通常、移動に足を用いている。すなわち、歩く、走るということが移動の基本になっている。仮想環境においても、歩くことによって移動を行うのが自然であると考えられる。歩行インタフェースを実現する方法としては、実世界での歩行を検出することが考えられる。しかし、実際に歩きまわる人間の動作をすべて取得するためには、動く範囲すべてをカバーするセンサが必要となる。また、対象となる仮想環境と同じ広さの空間が必要になる。この問題を解決するために、その場での歩行動作を可能とする方法が種々提案されている。

岩田らは、ローラースケートや滑るサンダルを用いた歩くインタフェースを実現している [7][8]。これらのインタフェースでは、ユーザは磁気位置センサと床面との接触を検

出するセンサを取り付けた「滑るサンダル」を履いて輪の形のセーフティフレームに入る。その中で移動する方向に歩くが、実際にはセーフティフレームに移動を制限されるので足が滑るだけで移動することはない。センサが歩く動作を検出し仮想空間内の移動に反映させるので、ユーザは歩く動作をすることで仮想空間内を歩くことができる。

また、岩田らはトレッドミルを用いた方法も提案している。トレッドミルは、ベルト機構によりユーザの歩行動作を相殺する手法であり、ユーザにアクチュエータを装着する必要がない。ベルトの駆動方式から大別してユーザの重力分力からできるパッシブ型である。トレッドミルの問題点に、歩行方向がベルト方向に限定される点が挙げられる。

歩行インタフェースは、足によるインタフェースの自然な形である。現実空間での歩行を仮想空間での歩行に直接的に結び付ける歩行インタフェースでは、日常歩くのと同じように仮想空間内を歩くことができ、説明しなくてもその使い方を理解することができる。

これらのシステムでは、その装置上に行動範囲が限られること、装置が大掛かりになることが問題である。多面スクリーン式の没入型ディスプレイと併用する場合、足下への映像提示の障害となる。パワーの大きな機器的制御が必要となる。例えば、走行時に静止する場合などベルトを踏ん張って止めなければならない。

● 傾けるインタフェース

身体を傾けることによるインタフェースがFairchildらによって提案されている[12]。これは、頭部に取り付けた磁気位置センサによって計測した頭と状態の位置と傾きの変位量を用いたもので、2種類の方法を提案している。一つは頭の位置の変位量を指数関数的に増幅して仮想空間内の変位量に割り当てたもので、わずかに頭を傾げるだけで遠くへ移動でき、もとの位置に頭を戻せば原点に戻ってくる。もう一つは、変位量を移動速度に割り当てたもので、身体を傾げることでその方向に進むことができる。傾ける量が多いほど早く移動し、頭をもとの位置に戻せば止まる。

問題として、頭部を動かして何かあたりを見る場合、頭部が示す方向を検知してナビゲーションコマンドと誤って解釈し、誤ったナビゲーションをしてしまう場合がある。

小林らは、体重の移動を検出して移動の方向や速度を入力する方法を提案している[2]。Josephらは、体を傾げることにより移動の方向や速度を入力する方法を提案している[17]。

傾けるインタフェースの問題として、移動方向が制限される。また、歩行動作ではないので直感的ではないといった問題点が考えられる。

● ゲーム機等のインタフェース

体感型ゲームやスポーツトレーナーのインタフェースにも手以外の部位を主に用いるものがある。例えばオートバイの車体を傾けるインタフェースや、スキーのインタフェース[13]等である。これらのインタフェースは、多くの場合、決められたコースを走る中で、種目特有の身体の動きを入力し、反力としてリアルな力触覚フィードバックを返すことを目的に設計されている。

- 音声のインタフェース

Philipらにより、音声を入力、認識し、指示を出すインタフェースも開発されている[15]。しかし、音声認識が利用できない不適切である、または望ましくない状況に対しては有効ではない。

2.5 まとめ

仮想都市内では、実世界の人間の動作と同じようにインタラクションすることが要求されており、少なくとも仮想都市内を自由に移動し、仮想物体を操作する必要があることを示した。特に都市設計のための仮想都市環境システムでは、都市内を体験するために移動する必要があり、安全性や快適性に問題があれば、設計物や置物の配置を変える必要がある。このような仮想都市環境システムでは、少なくとも仮想環境内を移動し、仮想物体をつかみ別の場所へ移動することが必要となる。ここでは、従来研究のインタフェースを説明し、これらの用途として不十分であることを明らかにした。

続く3章では、本論文での仮想都市内移動方法について述べる。4章では、仮想物体操作について述べる。

第3章 仮想都市環境における移動インタフェース

3.1 はじめに

我々人間は、いろいろな歩行速度で歩くことができる。平均的な歩行速度は、男性で約 4.8km/h(80m/min), 女性で 4.5km/h(75m/min) 程度である。何かをしながら歩く場合にはより速く 100m/min 以上になることもある。さらに高速なものに競歩がある。世界記録では 200m/min 以上の速度になる。しかし、一般的に 120m/min 程度を越えるとエネルギー効率のよい走行運動に切り替えるため、120m/min を大きく越えるような高速の歩行をすることは日常生活ではほとんどない。

このように、人間は目的や場面によって歩行速度を変えている。歩行動作を取り入れた移動インタフェースにおいても、実際の歩行と同じような感覚で速度制御ができれば操作性が理解しやすく、扱いやすいインタフェースが実現できると考えられる。

我々人間は通常、移動に足を用い、歩く、走るということが移動の基本になっている。歩くことによって移動を行うのが自然であると考えられる。実際に歩きまわるためには、対象となる仮想環境と同じ広さの空間が必要になるが、実世界におけるユーザの動けるスペースには限りがある。そこで、本研究では、より人間の歩行動作に近い足踏み動作を適用する。その場での足踏み動作を移動インタフェースに用いる利点としては、

- 人間は、自然空間内では主に足で移動するため、足に仮想都市環境内の移動のための操作を担わせるのは直感的でわかりやすい。
- 実際の歩行動作のトラッキングのように広い場所を必要としない。
- 踏み外す等の危険もない。
- 足に移動を担わせるため、手で他の操作が可能になる。

という点が挙げられる。

仮想空間内を自由に移動するには、速度制御と方向制御が必要になる。ここでは、実際の歩行動作の計測結果から、足踏み動作による仮想空間内の歩行速度制御と移動方向制御の手法について議論する。3.2 で実際の歩行動作の計測について述べる。3.2 の結果に基づいて 3.3 で歩行速度の推定を述べる。3.4 では、足踏み歩行での一步目の予測方法について説明する。3.5 では、長距離の移動のための拡張歩行方法を説明し、3.6 では、仮想都市映像の拡大縮小による拡張歩行方法について概説する。3.7 では、身体方向を利用した仮想空間内移動方向制御について説明する。3.8 では、まとめを述べる。

3.2 実際の歩行動作の計測

3.2.1 計測方法

人間は目的や場面によって歩行速度を変えている。足踏み歩行により、実際の歩行と同じような感覚で速度制御ができるように、人間が通常どのように歩行速度を変えているかを調べる。今回、実際の歩行動作の3次元位置情報を取得するために、磁気センサーである Motion Star を用いデータを取得した。取得したデータを分析するために、計算機 Onyx を用いて処理を行った。

図の 3.1 に Motion Star を示す。Motion Star はモーションキャプチャーシステムとして従来の Flock of Bird を基本に改良した製品である。

Motion Star は人体アニメーション、生体力学、CAD、フライトシュミレーション、医療用、テレロボティクス、動作研究、VR (仮想現実) その他、多様化したアプリケーションがある。レーザー (小型受信器) を頭、胴、手、脚等に装着したり、その他の動作物体に取付けてそれぞれの運動を測定することが可能である。以下に Motion Star の詳細を示す。



図 3.1: Motion Star

- 自由度：6 自由度 (位置、角度)
- 最大センサー数：18 センサー/人
- 計測可能 角度：±180 ° (アジマス、ロール) ±90 ° (エレベーション)
- 計測範囲：±10 フィート (3 m) の半球内位

- 置精度：0.3 インチ (7.6 mm) RMS (1.5 m 範囲) 0.6 インチ (15.2 mm) RMS (3 m 範囲)
- 角度精度：0.5 °RMS (1.5 m 範囲) 1.0 °RMSs (3.0 m 範囲)
- 位置分解能：0.03 インチ (0.76 mm) (1.5 m 範囲) 0.10 インチ (2.54 mm) (3.0 m 範囲)
- 角度分解能：0.1 ° (1.5 m 範囲) 0.2 ° (3.0 m 範囲)
- アップデートレート：最大 100 データ/秒
- 出力：X、Y、Z
- インターフェイス：Ethernet, RS232C

今回の実験では人間の歩行動作について、かかとの高さ、膝の高さ、股下の角度、ピッチ、歩幅、歩数、歩幅、歩行速度について調べた。計測は、約 7m の直線歩行路を設定し、磁気センサーである Motion Star を用いて、歩行中の両足のかかとの位置をサンプリング周波数 88.6Hz で測定した。

図 3.2 に歩行動作計測環境を示す。

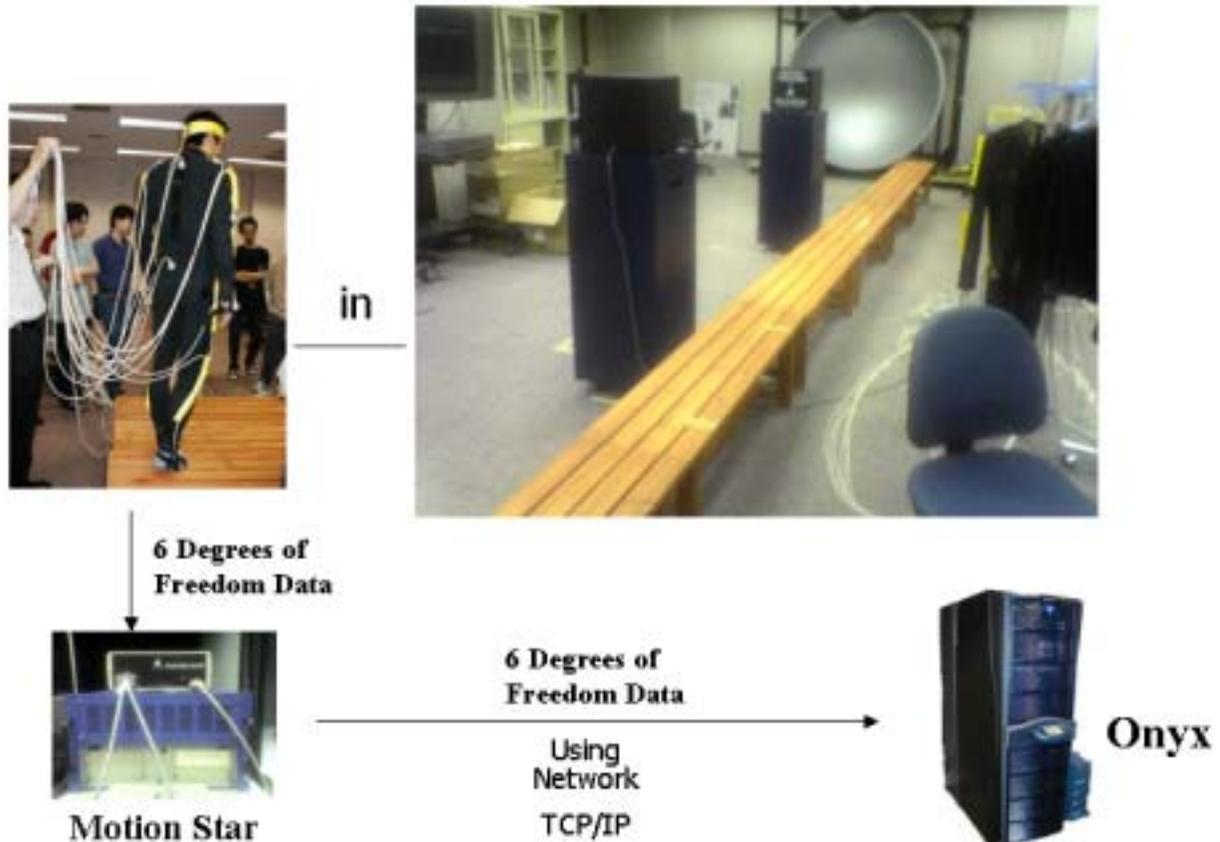


図 3.2: 歩行時のかかと位置計測結果

被験者は 25 才の成人男性 3 名である。実験では、遅く、普通に、早く、歩行するように

指示し、それぞれ4回の試行、合計36回の試行を測定した。被験者の自然な歩行のデータを得るために、あえて抽象的なおよその速度を指定した。歩き始めた直後と止まる直前では歩幅が小さくなることが予想されるため、実際にはスタート地点の2m手前から、ゴール地点の2m先まで歩いてもらい、初めてスタート地点を越えた着地点から、初めてゴール地点を越えた着地点までを妥当なデータとして測定した。

3.2.2 歩行動作の分析

- かかとの高さ

人は歩行時に、足を上げながら歩く。そこで、被験者の床からのかかとの高さに注目した。図3.3に被験者の歩行時のかかとの高さを示す。横軸は時間、縦軸は被験者のかかとの高さである。かかとの高さの上下運動から、被験者が歩行している様子が分かる。しかし、この結果からは、被験者の「遅い」歩行と「速い」歩行では、かかとの高さに大きな差がみられない。そのため、足踏み歩行での速度推定には不適であると判断した。

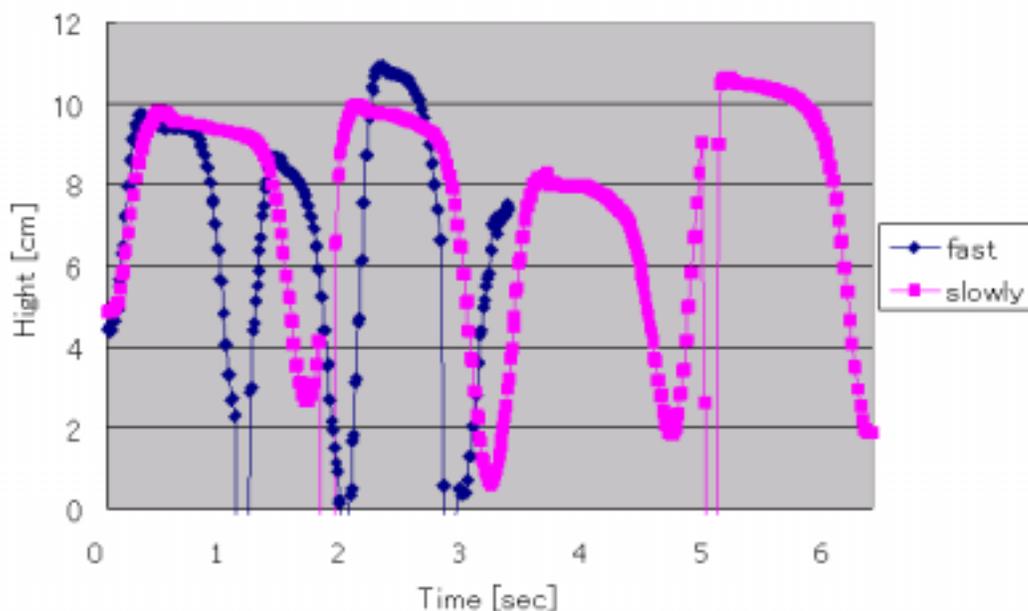


図 3.3: かかとの高さ

- 膝の高さ

人は歩行時に、足を上げながら歩く。そこで、被験者の床からの膝の高さに注目した。図3.4に被験者の歩行時の膝の高さを示す。横軸は時間、縦軸は被験者の膝の高さである。

この結果から、人の歩行時には膝の動きが複雑であることが分かる。そのため被験者の歩行の判断を行うことができない。そこで本研究では、足踏み歩行での速度推定には不適であると判断した。

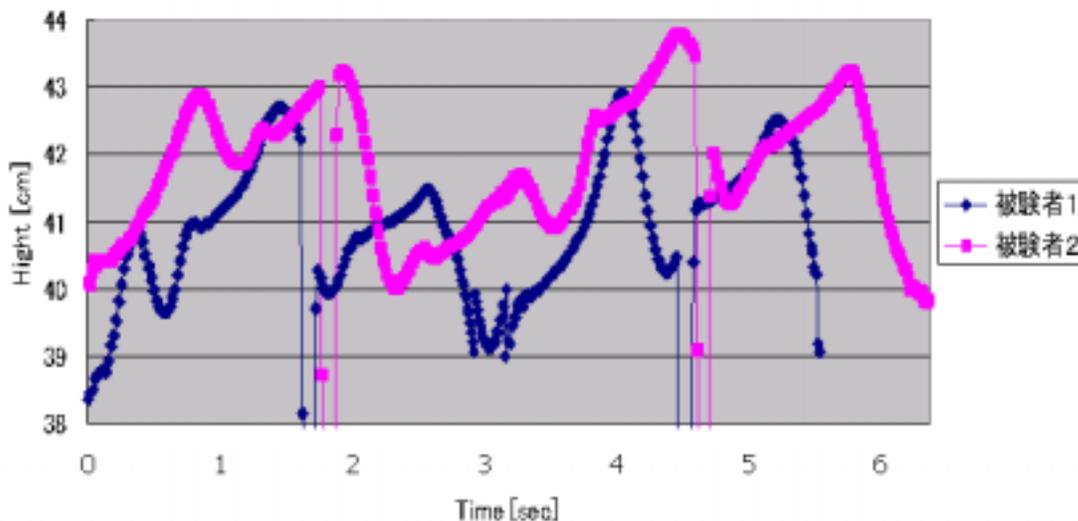


図 3.4: 膝の高さ

- 股下の角度

人は歩行時に、足を上げることで股下が開く。そこで、被験者の股下の角度に注目した。図 3.5 に示すように被験者の股下の部位から角度を抽出する。図 3.6 に被験者の股下の角度を示す。横軸は時間、縦軸は被験者の股下の角度である。股下の角度を $\cos \theta$ で示しているため、足を上げていない時には、 $\cos \theta$ が 1 になる。 $\cos \theta$ の変化から、被験者が歩行している様子が分かる。しかし、この結果からは、被験者の「遅い」歩行と「速い」歩行では、被験者の股下の角度に大きな差がみられない。そのため、足踏み歩行での速度推定には不適であると判断した。

- ピッチと速度

ピッチとは 1 秒間に歩く歩数と定義し、図 3.7 に歩行速度とピッチの関係を示す。縦軸は歩行速度で、横軸はピッチを示している。この結果から、歩行速度とピッチの間に比例関係があることがわかった。

- 歩幅と速度

歩幅とは足が離床してから着床する間の距離と定義し、図 3.8 にピッチと歩幅の関係を示した。図 3.8 を見ると、約 10cm 程度の幅で歩幅が一定に収まっており、ピッチに対して歩幅の変化が少ないことが認められる。

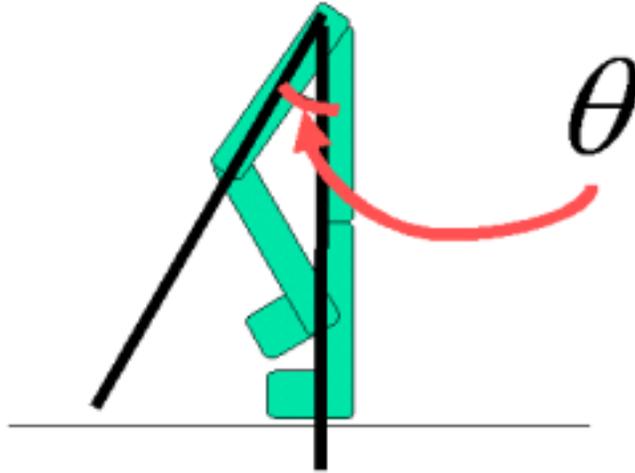


図 3.5: 股下の角度を抽出する部位

3.3 歩行速度の推定

移動の際に足踏み動作を行っても、実際に移動するわけではないので、足踏み歩行をした際の移動速度を推定する必要がある。図 3.7 に示したように、実際の歩行速度とピッチに比例関係が見られたため、本研究では、ピッチから足踏みによる移動速度を推定する手法を採用する。移動速度 V は歩幅 W とピッチ P の積によって $V = WP$ として与えられる。したがって、移動速度を制御するためには、どのように歩幅を決定するかについて検討する必要がある。前節の分析で、図 3.8 に示されるようにピッチに対して歩幅の変化が少なかった。そこで、本研究では、歩幅を一定と考え、 c [m] とした。図 3.8 の結果から 3 名の被験者の平均歩幅は、0.43m であった。そこで $c = 0.43$ m と設定した。

足踏み動作からピッチを求めるには、一步に要する時間を計測し、その逆数を求めればよい。足を上げた時点と足を下ろした時点を一歩とし、その周期を用いる。 n 歩目で足を上げた時刻を S_n 、足を下ろした時点での時刻を E_n 、その一步でのピッチを P_n とすると P_n は次式で表せる。

$$P_n = \frac{1}{E_n - S_n} \quad (3.1)$$

歩幅は定数 c とし、 $n + 1$ 歩目で進む速度 V_{n+1} を次式で求める。

$$V_{n+1} = cP_n \quad (3.2)$$

図 3.9 は足踏み歩行におけるピッチに対する位置の決定の様子を表したものである。横方向が時間の経過を表す。 t_0 から t_1 、 t_1 から t_2 と t_7 まで、それぞれが一步に要する時間を表している。一步に要する時間が長いとピッチが小さくなり、それに応じて速度が減速し

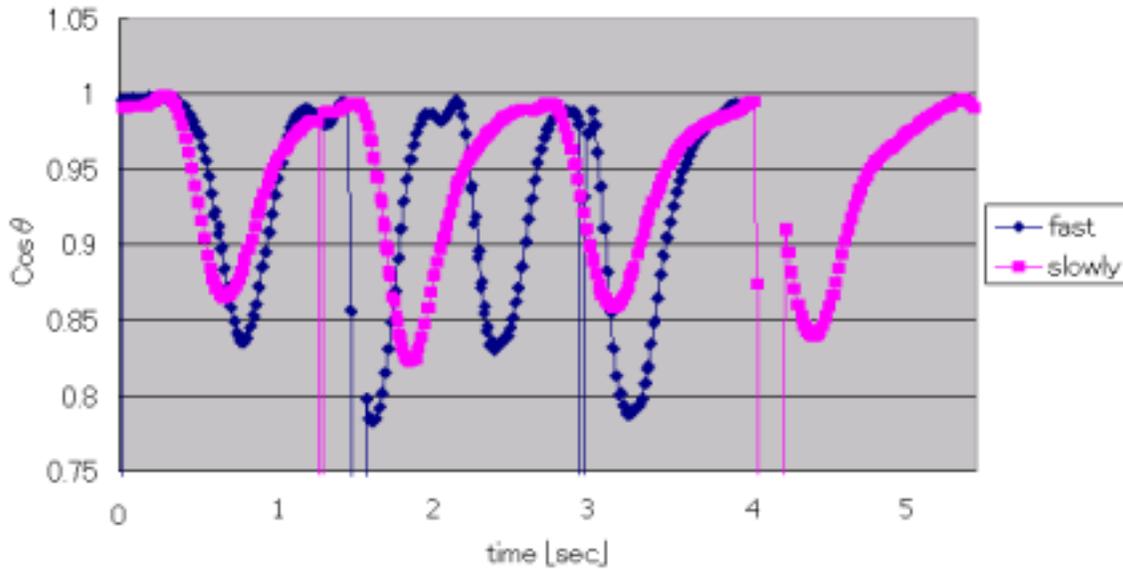


図 3.6: 股下の角度

進む距離も少なくなる。また一步に要する時間が短いとピッチが大きくなり速度が加速することにより進む距離が大きくなる。

3.4 一步目の予測

前節の(3.2)式では、次の一步目の速度を予測しているため、歩き始めの一步目で仮想空間を移動することができないという問題がある。人は足を上げた時点で歩き始めているが、(3.2)式を用いると、一步目の足を下ろした時点で速度が決定され、移動を始める。これによって、実際に移動したつもりなのに仮想空間が反応せずに違和感が生じることが分かった。このため、被験者の一步目のみ、あらかじめ予測した速度を与えることにした。実空間での歩行から、一步目の平均速度を算出したところ、平均0.16m/secであることが分かった。図 3.10 に一步目の速度を示す。この値を一步目の速度として用いることにした。

図 3.11 は足踏み歩行における一步目の速度を予測し、ピッチに対する位置の決定の様子を表したものである。横方向が時間の経過を表す。一步目から、足を上げた時点で速度が出るため、 t_0 の時点で、位置が進んでいる様子が分かる。

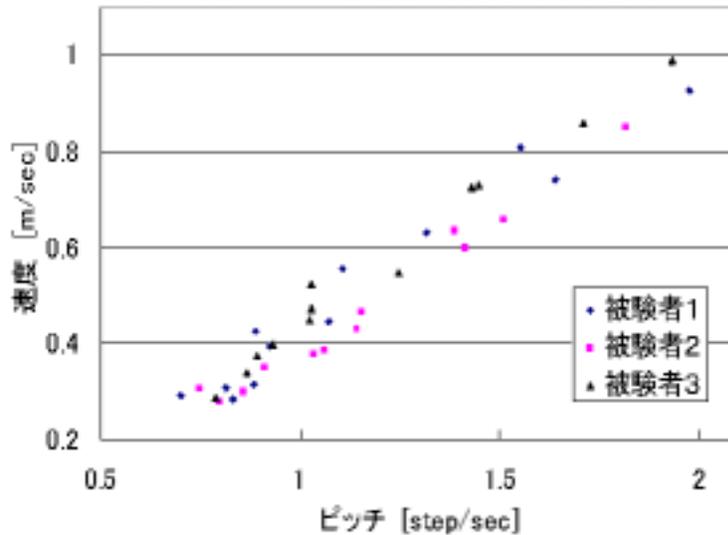


図 3.7: 歩行速度とピッチの関係

3.5 拡張歩行

第一に、インターフェースは操作が理解しやすく、使いやすいものでなければならない。この点で、歩行動作に適していると考えられる。一方、ウォークスルーでの長距離移動に要する時間を考えた場合、移動は短時間で行えることが望まれることがある。すなわち、高速で移動できることが必要となる。瞬時に目的の場所に移動する方法も考えられるが、自分の位置を見失う可能性がある。また、高速でしか移動できない場合、重要な目印を見過ごしてしまったり、目的の場所に停止することが困難になったりすることが考えられる。例えば仮想都市でのウォークスルーを考えた場合には、目標が遠くにあるときには自動車や電車のように高速で移動し、目標に近づいてきたらゆっくりと歩くような速度で移動するというように、速度を制御できることが重要である。しかも、その制御方法は、人間にとって理解しやすく、扱いやすく、大きな負担をかけないものでなければならない。したがって、使いやすいインターフェースとしては、低速から高速まで、速度を自在にかつ容易に変えることが必要である。

歩行動作を取り入れた移動インターフェースにおいて、実際の歩行での歩幅を適用することで実際の歩行と同じパフォーマンスを提供することは、必ずしも効果的ではない。

仮想環境においては、歩幅を任意に設定することができる。例えば、高ピッチの領域でも歩幅をピッチに比例させるようにすれば、実際の歩行より速度を上げることが可能になる。この場合には、低ピッチでは実際の歩行と同じような感覚で移動することができ、高ピッチでは実際には歩けないような速度で移動することができるようになる。あるいは、非常に大きな歩幅を考えれば、やはり高速に移動することができる。

本研究では、前節の歩行速度制御に加え、仮想環境であることの利点を生かした歩行制

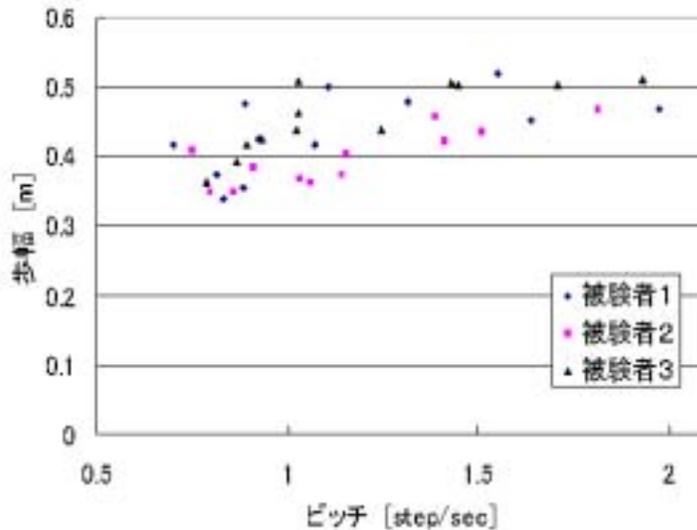


図 3.8: 歩幅とピッチの関係

御を実現する。大きな歩幅を設定すれば、高速に移動することができ、歩幅を 2 倍 3 倍 4 倍と変更することにより、 $V_{n+1} = (2 * c)P_n$, $V_{n+1} = (3 * c)P_n$, $V_{n+1} = (4 * c)P_n$, というように高速移動を可能な歩行システムを構築した。

3.6 仮想都市映像の拡大縮小による拡張歩行

隣の都市や国へ移動するといった場合など、前節でも述べたが、足踏み歩行のみで長い距離を移動する場合には疲労を伴うという問題がある。仮想環境においては、仮想都市映像を拡大縮小することができる。

本研究では、前節での拡張歩行に加え、仮想都市を拡大縮小することで容易に移動する方式も実現した。仮想都市を拡大縮小するために、CAVE 操作の標準インタフェースである 3DWand を用いる。拡大するにはボタン 1 を押し、縮小するにはボタン 2 を押すという操作を行い、自由に仮想都市を拡大縮小する機能を付加した。図 3.12 は等身大の仮想都市の映像を示している。図 3.13 は 3DWand によって仮想都市空間を縮小している様子を示している。図 3.13 のように 3DWand を手に持ちボタンを押すことで仮想都市空間を拡大縮小させることが可能である。

3.7 身体方向を利用した仮想空間内移動方向制御

仮想空間内で移動することを実現するには、速度の制御に加えて、移動方向の制御も必要になる。野間ら [19] によって提案された歩くインタフェースであるトレッドミルは、ベ

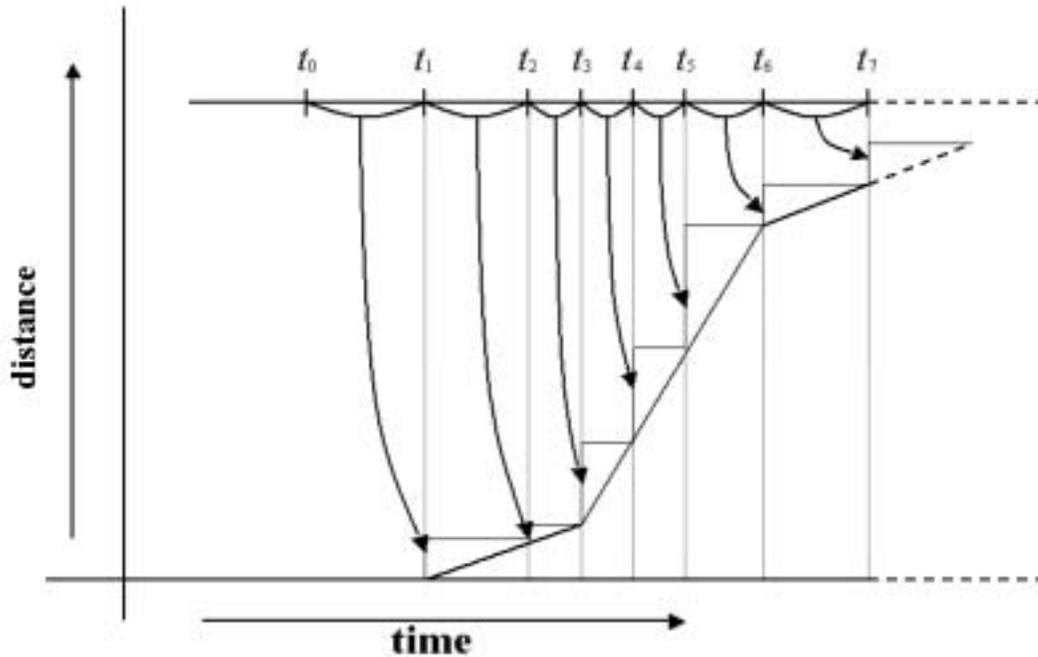


図 3.9: ピッチに対する位置の決定の様子

ルト方向に向きが限定され、仮想空間内での自由な移動方向の制御機能は有していない。本研究では、ユーザの身体方向を検出し、これを仮想空間内での移動方向に反映することで、任意の方向への移動可能な方式を構築する。移動方向を決定する方法として、体のいずれかの部位の方向を用いることが考えられる。頭部の向きを移動方向とする場合、一つの方向を見ながら他の方向に動くという一般的なかつ望ましい現実世界の移動操作を実施することが不可能であるということが、既に Joseph J.LaViola [17] らにより報告されている。

本研究では、図 3.14 に示すように、移動方向をユーザの胴体の向きで検出し、その向きに応じて仮想世界でのユーザの移動方向を決定することにした。胴体の向きの検出を行うために、ユーザの腰部の左右に 2 点の座標を取りつけた。2 点間の座標を結んだベクトル (\vec{a}) を求め、そこから垂直となる法線ベクトル (\vec{n}) の単位ベクトル (\vec{e}) を求め進行方向とした。センサーの座標を人間の左側を $M0(x0,y0)$ 、右側を $M1(x1,y1)$ とすると、

$$\vec{a} = (x1 - x0, y1 - y0) \quad (3.3)$$

$$\vec{e} = \frac{\vec{n}}{|\vec{n}|} = \frac{-(y1 - y0), x1 - x0}{\sqrt{(y1 - y0)^2 + (x1 - x0)^2}} \quad (3.4)$$

となりユーザの進行方向が計算される。

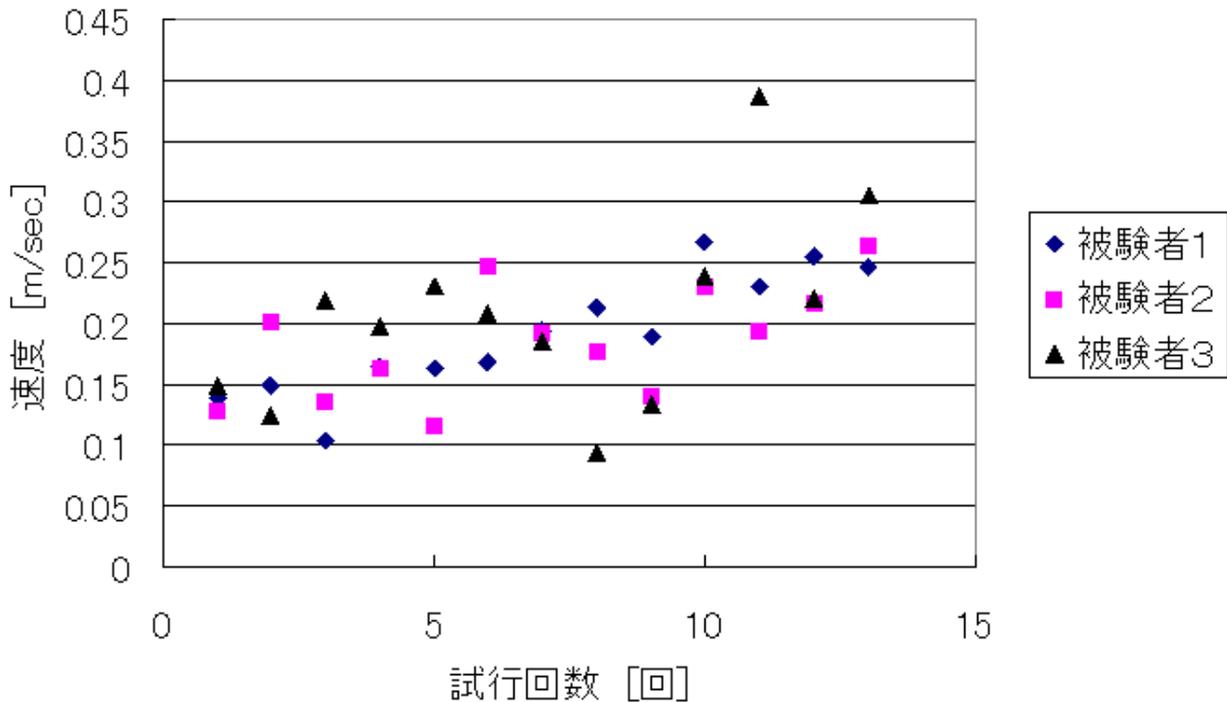


図 3.10: 一步目の歩幅

3.8 まとめ

本章では、人体に設置した特徴点の3次元座標から、仮想都市内の移動方法と仮想物体操作方法を提案した。

実際に仮想空間内を自由に移動するには、速度制御法と方向制御法が必要になる。そこで、モーションキャプチャによって得られる3次元位置データから、移動の際の速度制御および方向制御を提案した。仮想都市空間内の移動インタフェースには、人間の歩行動作に近い足踏み歩行を用いた。そこで、足踏み動作での速度制御のため、実際の歩行動作を調べた。その結果、歩行速度とピッチの間に比例関係が見られたため、本研究ではピッチから移動速度を推定する手法を採り、これを基に足踏み歩行により仮想空間を歩行するインタフェースを提案した。さらに、仮想都市環境であるという特徴を生かして、高速に移動する場合、歩幅を大きくすることで歩行速度を速くする機能も付加した。他の都市へ移動する長距離移動の場合は、仮想都市の縮尺を変えることで容易に移動する方式も実現した。

仮想都市内では、移動の他に、手でいろいろな操作を行う場面がある。第4章では、仮想物体操作について述べる。

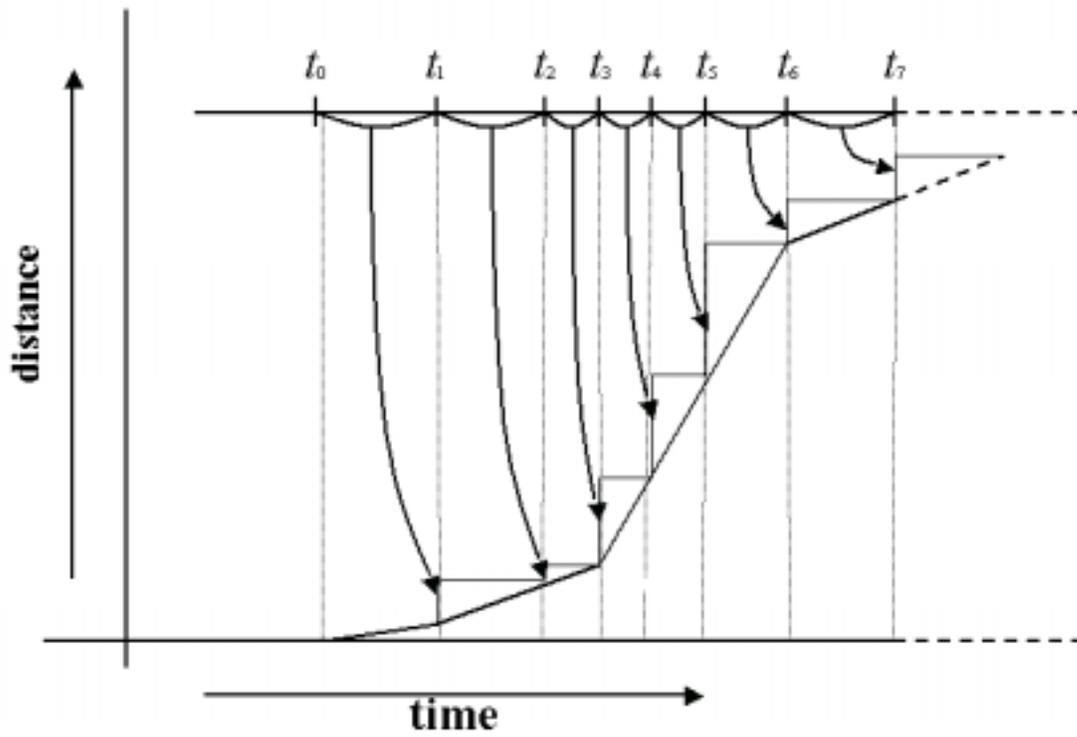


図 3.11: 一步目の予測した速度の様子



図 3.12: 縮小前の仮想都市映像



図 3.13: 仮想都市映像の縮小

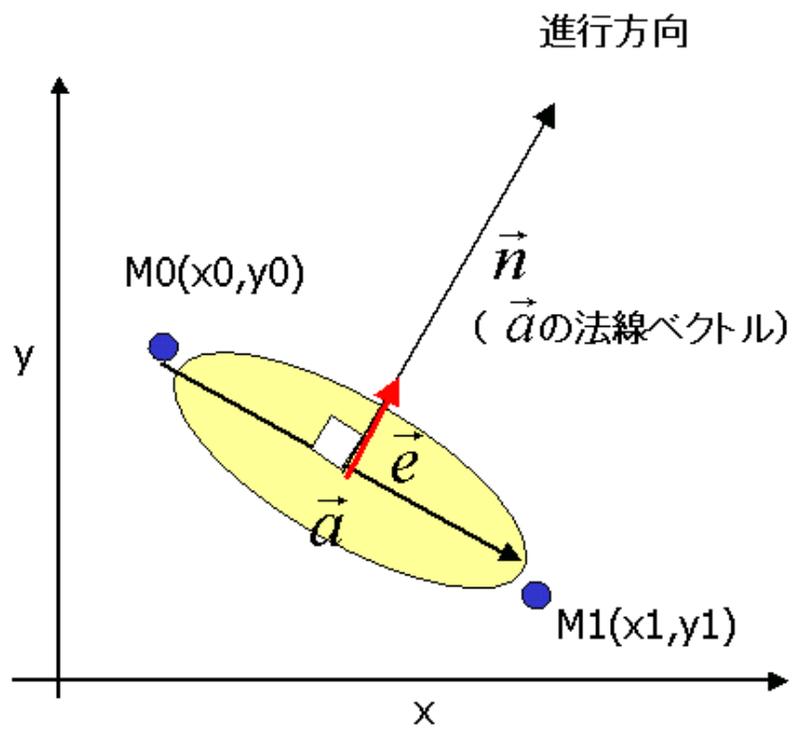


図 3.14: 移動方向の決定

第4章 仮想都市環境における仮想物体操作インタフェース

4.1 はじめに

仮想環境内では移動だけではなく、いろいろな操作を行う場面も考えられる。その操作の多くは手を用いることが想定される。仮想物体操作において、人間が実世界で行う動作で行うことは、作業性を考える上で重要な要素であり、人間が物体操作を認識する過程では、対象をつかむことによって直感的に把握することが必要である。

これまでの仮想空間インタラクションインタフェースとして、2章でも述べたがジョイスティックや3DWandなどのインタフェースが提案されてきた。しかし、これらのインタフェースは、仮想物体操作において、自然な動作はなく直感できではない。また、仮想空間内の移動と仮想物体の操作を全て行い、操作性の面で理解しにくいことが挙げられる。そこで、本研究では、人が実世界で行う動作のように、仮想物体を操作する手法を提案する。

仮想物体の操作法も、人間が実世界で行う動作に近い方が望ましい。本研究では、握るという動作により物体を選択する仮想物体移動法を提案する。ユーザの手首と手先にそれぞれ1つ座標を設置したモデルを考え、ベクトルの大きさを計算する。そのベクトルの大きさがある閾値を越えた場合に握ったと判定する。仮想物体と手の接触判定は、仮想オブジェクトを直方体の領域で取り囲み、この直方体の中に、腕先に設置した座標が入ることで接触していると判定する手法を採用した。

本章では、仮想物体操作システムについて述べる。以降、4.2では、仮想物体操作の過程を説明する。4.3では、まとめを述べる。

4.2 仮想物体操作の過程

仮想物体操作を行っている様子を図4.1に示す。

仮想物体を操作するには、仮想物体との接触判定を行う必要がある。図4.2に手を握った動作の判定方法を示し説明する。座標をユーザの手首と手先にそれぞれ1つ座標を設置し、それぞれの座標間のベクトルの大きさを計測する。ある閾値を設定し、手を握った場合に座標間のベクトルの大きさが、その閾値より小さければ手を握ったと判定する手法を採った。閾値は、キャリブレーションを行うことにより求め、通常の手を開き伸ばしてい



図 4.1: 仮想物体を操作している様子

る状態時と手を握った状態時の座標間のベクトルの大きさの中間値を適当な値として閾値に設定した。

仮想物体との接触判定方法を述べる。仮想オブジェクトを図 4.3 のように直方体の領域で取り囲み、この直方体の中に、腕先に設置した座標が入ることで接触判定を行う手法を採用した。直方体の領域の作成方法は、仮想オブジェクトの座標値の中で、最大の x, y, z 座標の点と最小の x, y, z 座標の値の 2 点を基準とし、直方体を作成している。

図 4.4 にセンサーが領域内に入っている状態を示す。直方体の基準点である 2 点から直方体に沿った方向へのベクトルをそれぞれ $\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4, \vec{v}_5, \vec{v}_6$ とする。基準点から座標までのベクトルを \vec{a}, \vec{b} とする。ベクトル $\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4, \vec{v}_5, \vec{v}_6$ とベクトル \vec{a}, \vec{b} との角度が全て 90 度以内であれば、座標が立方体の中に入っていることになる。この計算式は次式で表すことができる。

$$\frac{\vec{a} \cdot \vec{v}_1}{|\vec{a}| \cdot |\vec{v}_1|} \geq 0 \wedge \frac{\vec{a} \cdot \vec{v}_2}{|\vec{a}| \cdot |\vec{v}_2|} \geq 0 \wedge \frac{\vec{a} \cdot \vec{v}_3}{|\vec{a}| \cdot |\vec{v}_3|} \geq 0 \wedge \frac{\vec{b} \cdot \vec{v}_4}{|\vec{b}| \cdot |\vec{v}_4|} \geq 0 \wedge \frac{\vec{b} \cdot \vec{v}_5}{|\vec{b}| \cdot |\vec{v}_5|} \geq 0 \wedge \frac{\vec{b} \cdot \vec{v}_6}{|\vec{b}| \cdot |\vec{v}_6|} \geq 0 \quad (4.1)$$

図 4.5 に座標が領域内に入っていない状態を示す。

接触判定の後、仮想オブジェクトをローカル座標で移動、回転することにより仮想物体を任意に操作する。ローカル座標とは、物体独自に持つ座標系であり、対象となるオブジェクトに対して動的に決定される。それに対して、ワールド座標とは、シーン全体を表現する、絶対的な座標系で、3次元モデルに1つだけ存在する。仮想オブジェクトの移

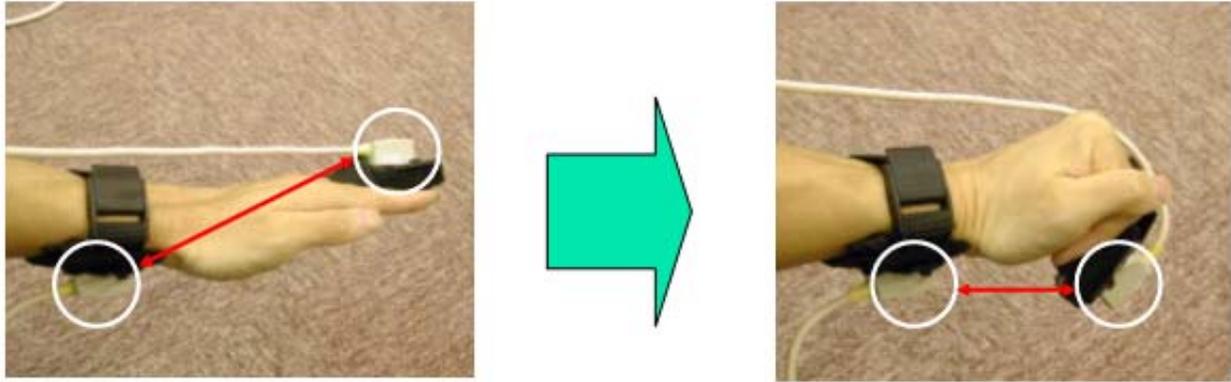


図 4.2: 握った動作の判定方法

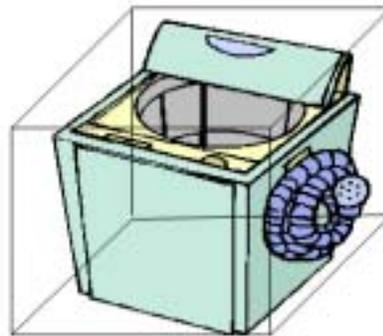


図 4.3: Maya から取り込んだオブジェクトを領域で囲んだ様子

動は、xyz 各軸方向の移動で指定する。まず、CAVE 内での手に身につけた座標を取得する。取得した座標の値は、ローカル座標での 3 次元位置情報が取得できる。このローカル座標を仮想世界のワールド座標に変換し、その座標のワールド座標と同様の座標に仮想オブジェクトを移動する。回転は任意の軸回りの回転で指定する。まず、CAVE 内で手に身につけた座標部分のオイラー角を取得する。このオイラー角を仮想世界でのオイラー角に変換し、その角度と同様にオブジェクトを回転させる。

4.3 まとめ

仮想都市内での仮想物体操作では、人間が実世界で行う動作のように、つかみ持ち運ぶことが必要である。仮想物体操作において、人間が実世界で行う動作で行うことは、作業性を考える上で重要な要素である。そのために、本研究ではつかむ動作を手を握る動

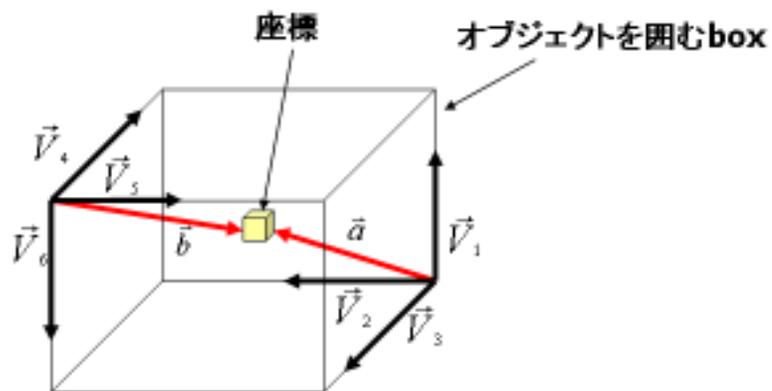


図 4.4: 座標が領域内に入っている状態

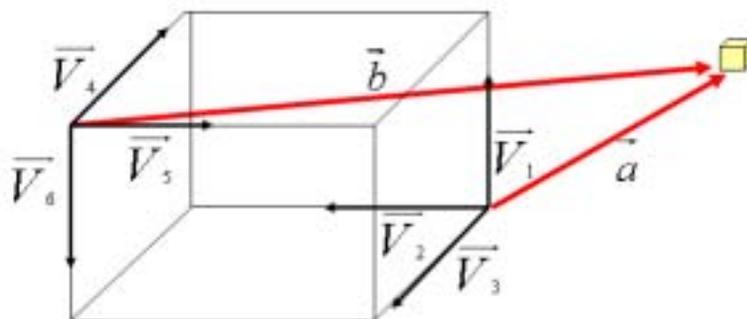


図 4.5: 座標が領域内に入っていない状態

作として採用した。また、手に付けた座標を基に仮想オブジェクトを操作する手法を提案した。

ユーザの手首と手先にそれぞれ1つ座標を設置し、座標間のベクトルの大きさがある閾値を越えた場合に握ったと判定する手法を提案した。仮想物体と手の接触判定は、仮想オブジェクトを直方体の領域で取り囲み、この直方体の中に、腕先に設置した座標が入ることで接触していると判定する手法を採用した。

3章では、仮想都市内移動方法について提案し、本章では、仮想物体操作について提案した。第5章では、これらの提案した手法を用いた仮想都市空間インタラクションシステムについて述べる。

第5章 仮想都市環境インタラクションシステム構成

5.1 はじめに

本章では、提案した手法を用いた仮想都市インタラクションシステムについて述べる。次に、実現した仮想都市環境システムの中で、従来研究であげた3DWANDと提案したインタフェースとの比較実験を行い、本インタフェースの有効性を示す。実験は、仮想都市内での移動実験を行った。仮想物体操作においては、3DWANDと本インタフェースとのアルゴリズムに差異がないため評価を行わなかった。

5.2では、システムに使用した入出力装置について説明する。5.3では、システム構成について概説する。5.4では移動インタフェースの速度制御と方向制御の実験について述べる。5.5では、3DWANDとの比較実験について述べる。5.6では、まとめを述べる。

5.2 入出力装置

5.2.1 Flock of birds

Flock of BirdsはDC(直流)磁界を使用した磁気センサーで、リアルタイムで三次元の位置と角度を検出することが可能である。小さな軽いセンサーを採用しており、高速な動きにも追従したトラッキングが可能である。図5.1にFlock of Birdsを示す。ステレオスペースデスク高速サンプリングセンサーは複数同時計測も可能で、それぞれのセンサーは144Hzで位置と角度を検出することが可能である。データの欠落なく、頭や体の関節の動きなどを計測できる。以下にその詳細を示す。

- 自由度：6 (位置及び角度)
- 有効範囲：半径4フィートの半球内
- 広範囲トランスミッター使用時：10フィート
- 角度有効範囲：アジマス, ロール 180°, エレベーション 90°
- 静的位置精度：0.07インチ RMS
- 静的角度精度：0.5°RMS
- 静的位置分解能：0.02インチ
- 静的角度分解能：0.1°

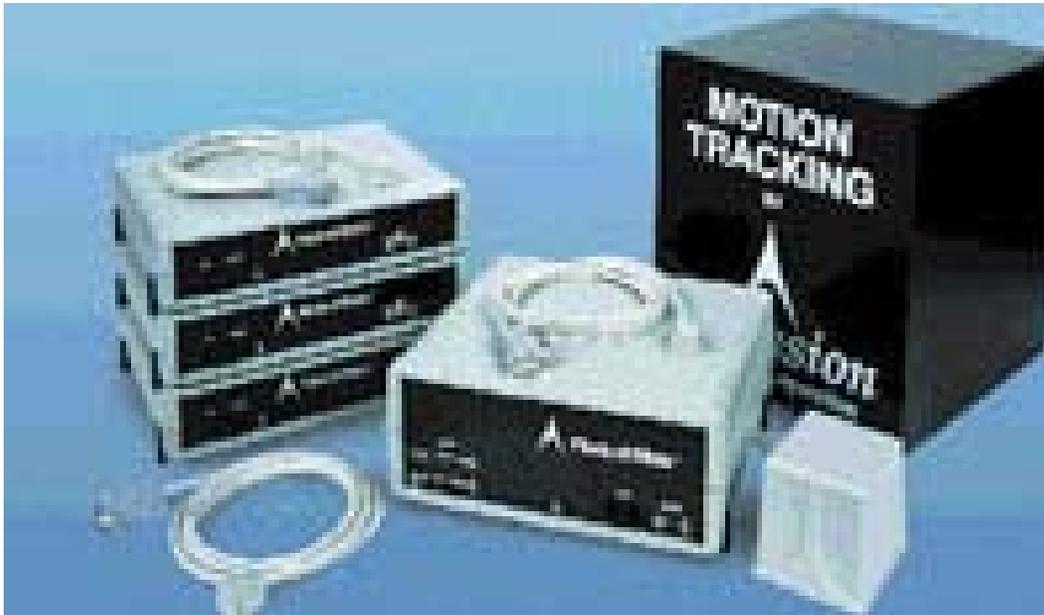


図 5.1: Flock of Birds

- データレート：最大 144 ポイント/秒
- 出力：X, Y, Z 位置及び角度 (オイラー角, マトリックス, 四元数)
- インタフェース：RS-232C 最大 115,200bps
- データ形式：バイナリ
- 標準トランスミッター：3.75 インチ立方体 10 フィートケーブル
- 広範囲トランスミッター (オプション)：12 インチ立方体、20 フィートケーブル
- センサー：1 インチ × 1 インチ × 0.8 インチ (長さ × 幅 × 高さ)、10 フィート / 35 フィートケーブル

5.2.2 CAVE

CAVE は、シカゴにあるイリノイ大学 Electronic Visualization Laboratory により、1980 年代より設計されたもので、立体的なバーチャルリアリティの表示方法である。CAVE という名前は、Cave Automatic Virtual Environment の頭文字を取ったもので、さらに哲学者が知覚、現実、幻想を探求するプラトンの Republic "洞窟の恩恵" からも由来している。CAVE は、四方を囲んだスクリーンと、サラウンド、サウンド、映写によるバーチャルリアリティ劇場から成っている。大きさが 10 フィート、10 フィート、9 フィートから成る直方体で、四方を完全に囲んだスクリーンに 3 次元コンピューターグラフィックスを映写し、幻想に没頭する世界を作り出せる (図 5.2)。軽量のステレオ眼鏡を全員が付け、頭と手の 2 つのトラッキングシステムで、遠近画像を矯正し、3D 入力素子の位置

と方位を鑑別する。CAVEでは、最大12名(パッシブ11名)までの視聴者がCAVEの中で仮想体験を共有することができる。ただし、1名の視聴者だけがアクティブ(対話が可能)で、他はパッシブな視聴者である。

没入型ディスプレイの利点の一つは、利用者の周囲を取り囲む大型スクリーンを配置することにより、広視野が得られることである。そこに提示される高臨場感仮想環境は、様々な分野での応用や研究が行われている。没入型ディスプレイが注目される理由には、高品質と安定性の良さがある。つまり、技術的には比較的成熟した映像投影を行っており、高性能プロジェクタの出現によって高解像度と高輝度が提供される。また、HMD(Head Mounted Display)と比較した場合、利用者が頭部回転運動を行ったときの映像変化が小さいので、安定した映像の提示が可能である。



図 5.2: CAVE

- 面数：4面構成
- スクリーンタイプ：ソフトスクリーン
- スクリーンサイズ：2.5×2.5 m
- プロジェクター：DLP プロジェクター (Mirage2000) 液晶プロジェクター選択可能
- 立体方式：液晶シャッター方式
- トラッキングセンサー：磁気式

- ライブラリー：CAVE.LIB
- Viewer：VRSCAPE（VRCO 社製）

5.2.3 シャッターメガネ

人間の目は左右に離れているため、それぞれには異なる像が写る。これを両眼視差といい、人間が立体を知覚するための重要な情報の一つである。CAVE は、この両眼視差を利用して立体的に表示する。スクリーンには右目用と左目用の画像が交互に高速に切り替わって表示される。これに液晶シャッターメガネ（図 5.3）が同期して、右目用の画像が表示されているときは右目のみ、左目用の画像が表示されているときは左目のみ見えるようにシャッターが切り替わる。左右目のそれぞれの違った視点で見た 2 つの映像がディスプレイモニタ上に 120Hz 以上でリフレッシュされる。結果として、ユーザーは左右それぞれの異なる映像を交互に 1 秒間に 60 回以上づつ見ることになる。この時、ユーザーの頭脳の中では 2 つの異なる映像が自然に両眼視差で見えるように融合され、完全なステレオ 3D 映像として再現される。液晶シャッターメガネの左右目用のシャッターの開閉は、赤外線エミッターからの赤外線信号によってコントロールされる。



図 5.3: シャッターグラス

5.2.4 立体視対応・超高精細 CRT プロジェクタ

図 5.4 に立体視対応・超高精細 CRT プロジェクタを示す。短残光 CRT を採用することで、144MHz クラスの高速シャッターメガネでも残像のない鮮明な立体映像を投影する。教育分野、シミュレーション、建築設計等の多種多様な 3D アプリケーションに活用可能である。



図 5.4: CRT

- 解像度 2500×2000 の超高精細映像
- コントラストモジュレーション機能
- ガンマ補正機能
- イメージシフティング機能
- ACON II / オートコンバージェンス機能
- ハイビジョン対応

5.2.5 Onyx3200

CAVE では3次元のコンピュータ・グラフィックスを表示し、ステレオ視のための2つの画像を同時に表示する。CAVE 内のユーザの頭の位置や向きが移動すると、それに対応した新しい画像を作成して投影する。SGI Onyx3200 (5.4) は、CAVE を使用するために必要な3次元ステレオコンピュータ・グラフィックスの製作、位置センサ情報に基づく位置および視点の移動処理などの各種の処理を行なうコンピュータである。特にCAVEでは高精度でリアルタイムにこれらの処理を行なう必要があるため、高速なプロセッサを複数個持つ、大変高速なコンピュータを使用している。さらに、コンピュータグラフィックスを特に高速に製作するためのハードウェア装置を使用している。

- プロセッサ : (MIPS R 12000 400MHz, 8MB レベル2 キャッシュ) ×4 個
- メモリ : 4GB、共有メモリ
- グラフィックス: InfiniteReality3 パイプライン ×2 個、256MB Texture Memory Raster Manager ×4 個、2-channel display generator ×2 個
- ディスプレイ : 24 インチ Super-wide Color Monitor



図 5.5: Onyx3200

- システムディスク : 18GB×2個
- ストレージディスク : SGI TP9100 Deskside Strage System(73GB ディスク ×2個)
×1個その他機器 : CD-ROM ドライブ、DAT
- ソフトウェア:IRIX 6.5 Advanced Workstation Environment with WorldView Japanese,
IRIX Performer,IRIX C Compiler/C++ Compiler

5.3 システム構成

人物動作を認識するためには各身体部位の3次元位置を計測することが必要であるが、3次元位置計測の機器としてはMotion Star と Flock of Birds がある。CAVE の内部という特殊な環境下において計測を行うことを想定し、2つの機器によりシステムを構成し、どちらの機器を使用するか検討した。

5.3.1 Motion Starを使用したシステム構成

図 5.6 に Motion Star を使用したシステム構成を示す。ユーザに磁気位置センサである Motion Star を取り付け、ユーザの動作の 3 次元位置データを取得する。処理装置として並列計算機である Onyx3200 を用い、視覚情報提示装置として CAVE を使用する。図 5.6 のようにユーザの背後に Motion Star の磁場発生装置 Tracker を設置し、ユーザが身につけたモーションスーツのセンサーによって、3 次元位置が計測される。また、CAVE 上部にも磁場発生装置 Tracker が取り付けられており、液晶シャッターメガネに取りつけられたセンサによって、ユーザの視線が計測される。

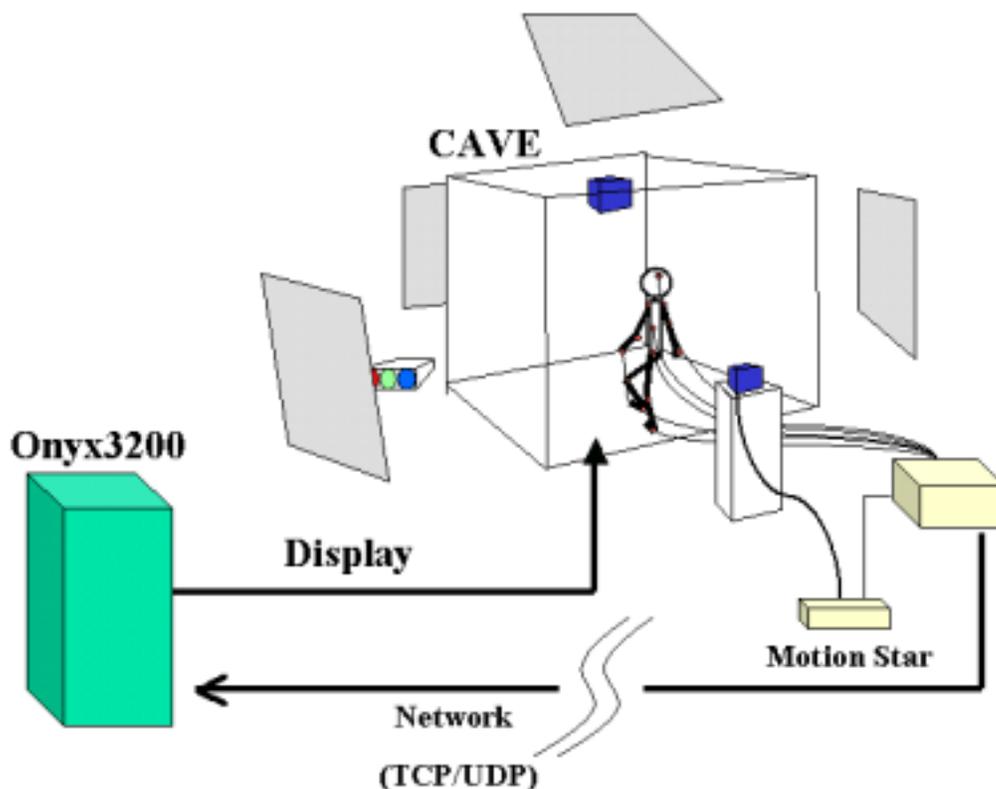


図 5.6: Motion Star を使用したシステム構成

Onyx3200 と Motion Star とのデータの受け渡しは、ネットワークを介して行う。ネットワークの通信速度は 100Mbps であり、Motion Star で取得できる全データ量は、1 秒間に 100Mbit 以下である。また、Onyx3200 と Motion Star のネットワーク間は他の通信による影響がないように専用の LAN で構築した。そのため、3 次元位置情報のデータ送信に対する遅延はなく、ほぼリアルタイムに 3 次元位置データを送信することが可能である。通信方式としては、TCP または、UDP での通信が可能なシステムを構築した。

TCP はコネクション型の通信であり、通信を始める時にはまず通信相手に送りたいデー

タがあることを伝え、通信相手との間で通信を行う準備が完了してからデータを送り始める。そのため信頼性のある通信を行うことが可能である。UDP はポート番号によるプロセスの識別とチェックサムによる誤り制御しか行わないため、信頼性を重視する通信サービスには適していない。しかし、高速で軽いプロトコルであるという利点がある。今回のシステム構成では、3次元位置データを正確にかつ、リアルタイムに得る必要がある。そこで、TCP とUDP で共に遅延に対する検証を行ったが両者とも通信の遅延に大きな差がなく、ほぼリアルタイムで3次元データを取得することができた。そこで、今回は、3次元位置データを正確に取得するため信頼性のある TCP を用いることにした。

しかし、このシステム構成で、動作検証をしたところ3次元位置データを正確に取得していないことが分かった。これは、Motion Star の磁気発生装置 Tracker と CAVE 上部にある磁気発生装置 Tracker の周波数が同じであり、共に干渉を起こし、3次元位置データを正確に取得することが困難であることがわかった。そのため、お互いの周波数を変えて検証を行ったが、前後 10cm 程度の3次元位置データの乱れがあることが分かった。

そこで、Motion Star を使用したシステムでは、Motion Star の磁場発生装置 Tracker と CAVE の磁場発生装置 Tracker が発生する磁場が少なからずとも干渉するため、本検討で、Motion Star を使用することは有効ではないと判断した。

5.3.2 Flock of Birds を使用したシステム構成

図 5.7 に仮想空間システムの概要を示す。ユーザの左右のかかとと腰部に磁気位置センサである flock of birds を取り付け、ユーザの動作の3次元位置データを取得する。処理装置として並列計算機である Onyx3200 を用い、視覚情報提示装置として CAVE を使用する。CAVE は4面（前、下、左、右）が2m×2m の大型ディスプレイに囲まれた空間に、任意の映像を立体表示させて、仮想都市空間を体験できる没入型ディスプレイである。CAVE 上部には磁場発生装置が取り付けられており、ユーザが身につけたモーションスーツと液晶シャッターメガネに取りつけられたセンサによって、センサの位置とユーザの視線が計測される。

Flock of Bird を用いたシステム構成では、Motion Star を用いた場合の干渉問題がなく、正確に3次元位置データを取得することができた。Motion Star は独自の Tracker を使用しなければならない。しかし、Flock of Birds の場合は、CAVE 上部の磁場発生装置 Tracker を使い、動作することが可能である。Onyx3200 のデバイスドライバを変更することにより、Flock of Birds との接続が可能である。Motion Star を用いず CAVE の磁場発生装置 Tracker のみの下に flock of birds のセンサーを1個から多数個に増やしこれらセンサーを身体に装着することで身体の動きを検出することが有効であると判断し、本研究では、Flock of Birds を用いることにより、人物動作の3次元位置を計測することにした。

図 5.8 は CAVE の中でモーションキャプチャを用いた歩行動作インタフェースにより仮想空間を移動している様子を示している。

5.3.3 CAVE への仮想都市の表示

図 5.9 に仮想都市を表示する流れを示す。3D アプリケーションソフトウェアを用い仮想都市オブジェクトを生成する。今回は、Alias Wavefront 社が開発した 3D アプリケーションソフトウェア Maya を使用した。Maya は OBJ データフォーマットにエクスポートする機能があり、この機能を利用し、生成した仮想オブジェクトを OBJ データフォーマットエクスポートした。データフォーマットをテキストエディタで開いたものを図 5.10 に示す。この OBJ データフォーマットには生成した仮想都市オブジェクトの頂点データなどが書き込まれている。この OBJ データフォーマットをオープンし、中に書かれたデータをリードして物体描画のための変数にセットすることで CAVE への表示を行っている。

5.3.4 仮想都市の表示のためのシステム内のプロセス

図 5.11 に仮想都市を表示するためのシステム内のプロセスの様子を示す。

CAVE は起動すると Application Initialaize プロセスで適切な透視投影変換、計算、ネットワーク、初期化等を行う。その後 CAVE Init プロセスで、CAVE システムに必要なプロセスを並行して実行する。Render Loop は、ディスプレイするためのプロセスである。CAVE のスクリーンは 4 つあり、1 つの Render プロセスで 2 つのスクリーンへの描画を行っている。また、3D 映像を提示するために左右それぞれの Left stereo ,Right stereo プロセスに分かれている。Tracking Loop は、センサーから取得した 3 次元位置情報を共有メモリに蓄積するプロセスである。Walk through Loop は、足踏み歩行により仮想都市空間内を移動計算を行うプロセスである。このように、CAVE アプリケーションは、マルチプロセスである。このため、共有メモリを用いることでプロセス間通信を可能にした。Maya から取り込んだ OBJ データフォーマットは、共有メモリに保存し、二つの Render プロセスで同期することで同じ OBJ フォーマットを読み込んで描画している。

5.3.5 背後の仮想空間へ移動する場合

CAVE には、背後のスクリーンがないという問題がある。そこで、背後の仮想空間の映像を前方に映し出すことで背後に存在する仮想空間へも移動することができるシステムを構築する必要がある。

しかし、背後の画像を急に前方に表示されるとユーザは自分が仮想空間上のどこにいるのか分からなくなる。人間の感覚の 80 % 以上を支配するのは視覚情報である。仮想空間を回転させながら背後の映像を前方に持ってくると人は、映像を見ながら仮想空間の中でどの位置にいるかを認識することができる。そこで、本研究では、仮想空間を回転されることで背後に存在する仮想空間への移動を行うことにした。

背後の仮想空間を回転させるために、CAVE 操作の標準インタフェースである 3DWand を用いる。右に仮想空間を回転するには右ボタンを押し、左に仮想空間を回転されるには

左ボタンを押すという操作を行い、自由に仮想空間を回転させる機能を付加した。

5.3.6 CAVEのスクリーンへの接触問題

人は、仮想空間内での没入感により、足踏み動作によりCAVE内で実際に移動する場合が想定される。その場合にCAVEのスクリーンなどに接触する危険がある。この問題点对して、本システムでは、被験者がCAVE内の中心点から半径50cm以上の範囲を越えた場合、足踏み歩行を行っても仮想空間内での移動を一時停止することにした。被験者が再び領域内へ戻ればシステムは再起動する。このことにより被験者が領域範囲内から出た場合は、システムを一時停止することで、スクリーンへの接触を回避した。図5.12はシステム動作領域範囲を表したものである。

5.4 移動インタフェースの速度制御と方向制御の評価実験

仮想空間内で、ユーザが任意に速度制御や方向制御ができていることを実験により示す。提案手法が実際の歩行と同じような感覚で速度制御、方向転換ができれば理解しやすく、扱いやすいインタフェースとして実現されたものと考えられる。提案手法によってユーザが任意に歩行速度を変更可能であること、及び方向転換が可能なことを実験により示した。

5.4.1 足踏み歩行による目的地までの速度制御

ユーザが思い通りに目的地まで速度制御をしているかを調べるために実験を行った。実験に用いた仮想環境は、目的地となるゴールを前方20mの位置に直径1mの円を地面に設置し、その中に被験者が入ることでゴールとした。実験のタスクは、スタートからゴールまでできるかぎり速く移動するという、正確にゴールに停止するというものである。

ピッチの検出には、被験者のかかとに磁気センサーを付け、センサーの初期位置から1cm上に上がったところを閾値とし、その閾値よりセンサーが上がっていれば足を上げたとして判別する。磁気センサーの計測誤差は1mmであり、足踏み動作以外の動きによる誤動作の範囲を考慮し、閾値を1cmとした。こうすることで、ほぼリアルタイムで足を上げた時刻を検出することができる。3人の被験者に対し、それぞれの条件で2回ずつの試行を行った。以下に結果を示す。

図5.13、図5.14、図5.15はそれぞれの被験者の速度の時間変化を示したものである。被験者1の結果に注目してみると、最初の2秒程度までは速度を0.8m/sec位まで上げている。ゴールに近づくまでは0.8m/sec前後のほぼ一定の速度を維持している。そして、ゴールに近くになって速度を大幅に下げ、最終的には、0.5m/secまで下げている。また、他の被験者も同様な速度変化の傾向が見られた。ここから、ピッチを変えることで速度を

変化させ、速度を調節していることがわかる。

5.4.2 ターゲット追跡のための足踏み歩行による速度制御

ユーザがある動くターゲットを追跡することができれば、思い通りに速度制御が可能であることが言える。そこで、あるターゲットを被験者が同じ速度で追いかけるタスクを課した。追跡物は直径 1m の円を移動させ、円の中心点にマーカーを示し、被験者はマーカーの上方に合わせ移動するというものである。追跡物の動きはシステムによって管理されており、実験開始後 5 秒後に追跡物が 0.5m/sec で移動を開始し、20 秒経過した時点で 0.8m/sec まで加速する。実験開始から 30 秒経過後に再び 0.5m/sec まで減速し、45 秒経過した時点で停止する。被験者には、できるだけ同じ速度でこの追跡物を追いかける様に指示をした。結果を以下に示す。

図 5.16、図 5.17、図 5.18 は追跡物と被験者の歩行速度を示したものである。図 5.19、図 5.20、図 5.21 は追跡物と被験者との誤差を示したものである。実験開始 5 秒後の 0、15、30、45 秒後は、追跡物が急に速度を変化するため被験者と追跡物の誤差が大きくなる。しかし、その後は追跡物を追従し、被験者の歩行位置は設定位置を中心に前後 30cm 程度の範囲内に保持されている事がわかる。被験者はマーカを見ながら移動するが、追跡のしやすさからマーカを足下に見るのではなく約 20cm 程度前方にマーカを見ながら移動を行っている。この事を考慮すると誤差は約 30cm 程度あるが、歩行位置がほぼ目標位置に保持されているといえる。このことは、追跡物と被験者の移動速度と等しいとみなせ、追跡物の移動速度に合わせて移動速度を変化できていることが分かる。

5.4.3 足踏み歩行による目的地までの方向制御

ユーザが思い通りの目的地に移動可能である事を示すために実験を行った。実験に用いた仮想環境は、目的地となるゴールを前方 20m、左方 5m の位置に直径 1m の円を地面に設置し、その中に被験者が入ることでゴールとした。実験のタスクは、スタートからゴールまでできるかぎり速く歩き、できるだけ正確にゴールに停止するというものである。3 人の被験者に対し、それぞれの条件で 2 回ずつの試行を行った。

目的地までに達する被験者の経路について調べる。図 5.22 は被験者のゴールまでの経路を表したものである。目的地に近寄るために、ユーザが進行方向をスムーズに変えていることがわかる。仮想環境内でこのような自然な動きができるのは、本インタフェースの効用であると考えられる。

5.4.4 足踏み歩行による経路上の方向制御

ユーザがある経路を外れずに移動することができれば、方向制御ができていくということが言える。そこで、以下の実験を行った。実験に用いた仮想環境は、スタートから前方

に直線 20m、90 度右方に曲がり直線 5m、90 度左方に曲がり直線 20m の経路を示しそこをゴールとした。実験のタスクは、経路から外れないように移動し、正確にゴールするというものである。上記のタスクを 2 回試行実施し、被験者の移動経路と停止位置誤差を計測した。

図 5.23 に、被験者が目的地までに達する経路を示す。被験者は、直線では大きく外れることなく経路に沿って移動している。コーナーでは、ユーザが進行方向をスムーズに変えていることがわかる。この結果から、ユーザが任意に移動ができていることがわかる。仮想環境内でこのように任意な動きができるのは、本インタフェースの特徴である。

図 5.24 に実験での停止位置誤差の結果を示す。被験者の停止位置誤差はほぼ 20cm 以内に収まっている。足踏み歩行では、一步前のピッチを利用しているため、被験者の思った通りの移動速度と誤差が生じる。そのために多少の停止位置誤差が生じる。また実験の試行内で 5cm 以内に停止しているものも見られる。仮想都市内での移動を考えると十分であると考えられる。

5.5 3DWAND との比較

前節までに、速度制御、方向制御、停止位置誤差を独立で実験を行った。本節では、全ての要素を含めた実験を行う。また、従来研究で挙げた 3DWANS との比較を行い、提案したシステムの有効性を示した。

5.5.1 3DWAND との比較実験

あるターゲットを被験者が追いかけるタスクを課した。ターゲットの後方 2m の位置にマーカーを示し、被験者はマーカーの上方に合わせ移動するというものである。マーカーの動きはシステムによって管理されており、実験開始後 5 秒後に追跡物が 0.5m/sec で移動を開始し、ゴールまで 0.8m/sec で等加速する。ターゲットが通った後の道には、被験者が通るべき経路が示される。ターゲットの動きは 90 度のコーナーを 4 回曲がりゴールするというものである。また実験前にターゲットの動きは被験者には知らせていない。被験者には、できるだけ同じ速度でこの追跡物を追いかけ、経路をはずれずに、正確にゴールに停止する様に指示をした。

3 人の被験者に対し、それぞれの条件で 2 回ずつの試行を行った。

5.5.2 3DWAND の操作方法

図 5.25 に 3DWAND の構成を示す。3DWAND のジョイスティックのつまみを前後左右に押すことで対応する方向に移動することができる。進みたい場合には、Y 方向に押す。押す強さにより速度を制御することができる。曲がりたい場合には、仮想空間を回転させ

る。回転させるには、X方向にジョイスティックを押すことにより仮想空間を回転させることができる。回転する速度も押す強さにより制御できる。

5.5.3 3DWAND との比較実験結果

図 5.26、図 5.27、図 5.28 は、本システムを用いた時の追跡物と被験者との位置誤差を示したものである。図 5.29、図 5.30、図 5.31 は、3DWAND を用いた時の追跡物と被験者との位置誤差を示したものである。

被験者 1 の結果である図 5.26 と図 5.29 で比較する。

本システムでは、被験者の歩行位置は設定位置を中心に前後 30cm 程度の範囲内に保持されている事がわかる。5.5 節でも述べたが、被験者はマーカを見ながら移動するが、追跡のしやすさからマーカを足下に見るのではなく約 20cm 程度前方にマーカを見ながら移動を行っている。この事を考慮すると誤差は約 30cm 程度あるが、歩行位置がほぼ目標位置に保持されているといえる。しかし、3DWAND の場合では 3 箇所位置誤差が 30cm を越えている部分があり、それらは 3 つのカーブを曲がる地点にあたる。この理由を説明する。3DWAND で曲がろうする場合には、X 方向にジョイスティックを曲げなければならない。その場合、速度を制御するための Y 方向にジョイスティック押すことができない。そのために、速度制御と方向制御を思った通りに制御できないために、追跡物との誤差が大きく出てしまう。その様子が 3DWAND を用いた時の経路を示した図 5.32 にも示されている。

図 5.32、図 5.33、図 5.34 は本システムを用いたときの経路を示したものである。図 5.35 図 5.36 図 5.37 は 3DWAND を用いた時の経路を示したものである。これを見ると本システムを用いた場合には、被験者は直線では大きく外れることなく経路に沿って移動している。コーナーでは、ユーザが進行方向をスムーズに変えていることがわかる。しかし、3DWAND の場合には、コーナーの部分で大きくコースから外れていることがわかる。これは、前にも述べたように 3DWAND の操作に限界があるため速度制御と方向制御を同時に制御できないことが原因である。

次に停止位置誤差について評価する。図 5.38 は本システムと 3DWAND の停止位置誤差を示したものである。この結果から本システムの停止位置誤差は、18cm 以下である。仮想都市内での移動時から停止することを考えると十分であるといえる。3DWAND の場合には、18cm を越えて停止していることがわかる。これは、速度制御、方向制御ができていないために正確に停止できないのが原因である。

この結果から、本システムでは、ユーザが 3DWAND より任意に移動ができていることがわかる。仮想環境内でこのように任意な動きができるのは、本インタフェースの特徴である。

5.6 まとめ

本章では、提案した手法を用いた仮想都市インタラクションシステムについて述べた。また、本インタフェースの有効性を示すために、従来のインタフェースである3DWANDとの比較実験を行い、確認した。

提案した手法を用いた仮想都市インタラクションシステムについて述べる。インタラクションのために、モーションキャプチャを用いるとによって、人間動作の3次元座標データを正確に取得し、人体の動きに応じて仮想空間と直感的にインタラクションする。出力ディスプレイ装置としては、没入型ディスプレイ CAVE を用いる。この CAVE 内で、モーションキャプチャを身につけたユーザは実世界での動作と同様の動作で仮想都市空間内を移動し、物体を操作することが可能となる。

次に、実現した仮想都市環境システムの中で、本インタフェースと従来のインタラクションインタフェースである3DWANDとの比較実験を行った。実験は、あるマーカーを提示し、できるだけ同じ速度でこのマーカーを追いかけるというタスクを課し、その時のマーカーと被験者の位置誤差を測定する。ターゲットの動きは、等加速し90度のコーナーを3回曲がりゴールするというものである。実験の結果、本システムでは、被験者の歩行位置は設定位置を中心に前後30cm程度の範囲内に保持されている事が確認された。被験者はマーカーを見ながら移動するが、追跡のしやすさからマーカーを足下に見るのではなく約20cm程度前方にマーカーを見ながら移動を行っている。この事を考慮すると誤差は約30cm程度あるが、歩行位置がほぼ目標位置に保持されているといえる。一方、3DWANDの場合では、30cmを越えている部分があり、それらは3つのカーブを曲がる地点にあたる。これは、3DWANDの操作に限界があるため速度制御と方向制御を同時に制御できないことが原因であることがわかった。

この結果から、本システムでは、ユーザが3DWANDより任意に移動ができていることが確認された。

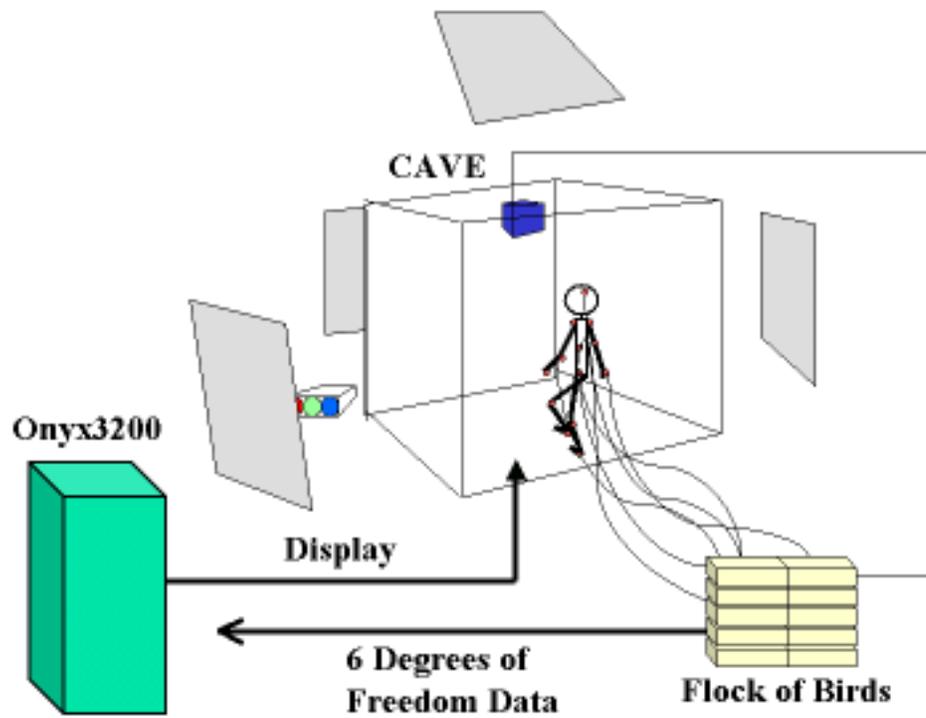
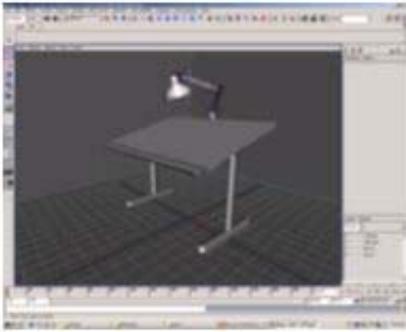


図 5.7: 仮想都市空間と歩行動作システム構成



図 5.8: 歩行動作インターフェースによる CAVE に構築された仮想都市空間内での移動の様子



Maya

export

```

building-1.obj (HIV) - GVIM
ファイル(F) 編集(E) ツール(T) シンタックス(S) ヘルプ(H)
v -6.111111 -0.091358 4.000000
v 6.111111 -0.091358 4.000000
v -6.111111 0.000000 4.000000
v 6.111111 0.000000 4.000000
v -6.111111 0.091358 4.000000
v 6.111111 0.091358 4.000000
v -6.111111 0.091358 -4.000000
v 6.111111 0.091358 -4.000000
v -6.111111 0.000000 -4.000000
v 6.111111 0.000000 -4.000000

```

OBJ Data Format

read



CAVE

Display



図 5.9: Maya から仮想オブジェクトを取り込む流れ

```

v -6.111111 -0.091358 4.000000
v 6.111111 -0.091358 4.000000
v -6.111111 0.000000 4.000000
v 6.111111 0.000000 4.000000
v -6.111111 0.091358 4.000000
v 6.111111 0.091358 4.000000
v -6.111111 0.091358 -4.000000
v 6.111111 0.091358 -4.000000
v -6.111111 0.000000 -4.000000
v 6.111111 0.000000 -4.000000
v -6.111111 -0.091358 -4.000000
v 6.111111 -0.091358 -4.000000

```

図 5.10: OBJ データフォーマット

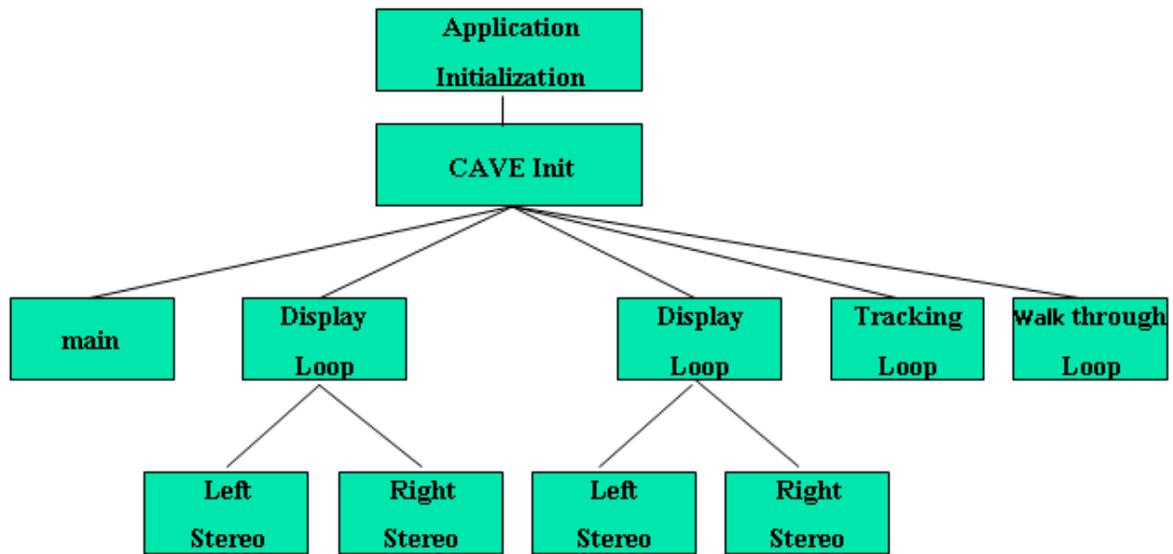


図 5.11: CAVE システムでのプロセスの様子

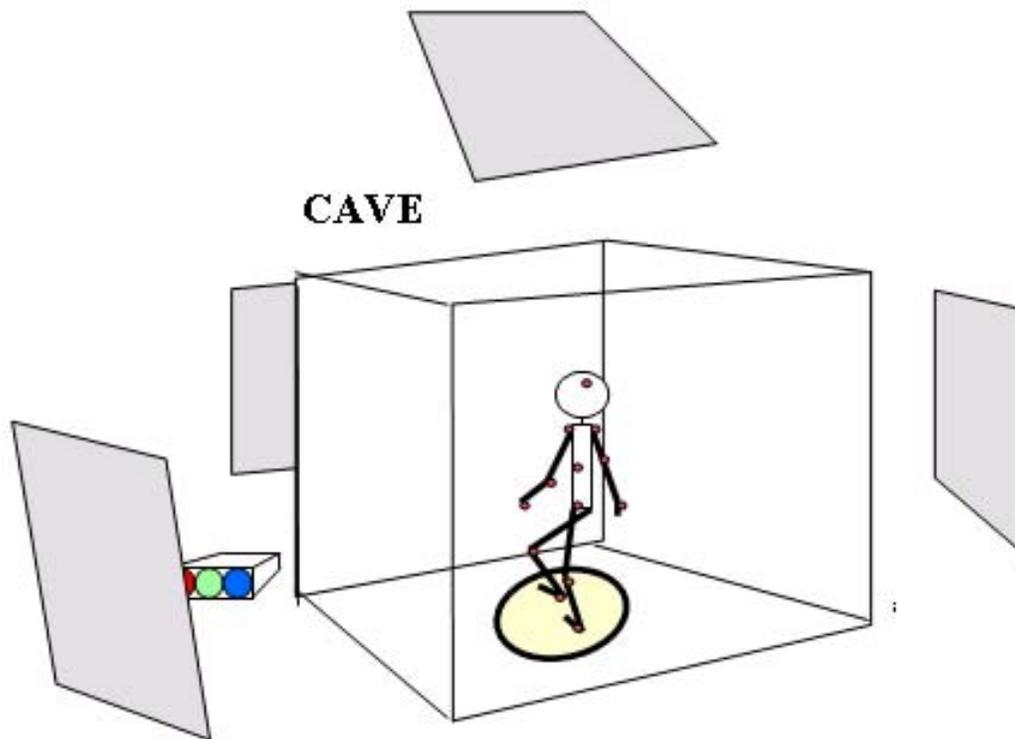


図 5.12: システム動作領域範囲

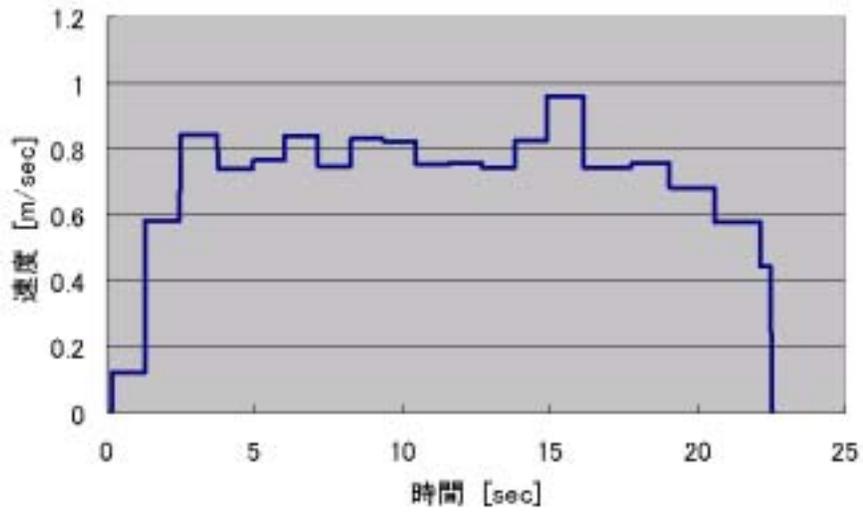


図 5.13: 被験者 1 の速度の時間変化

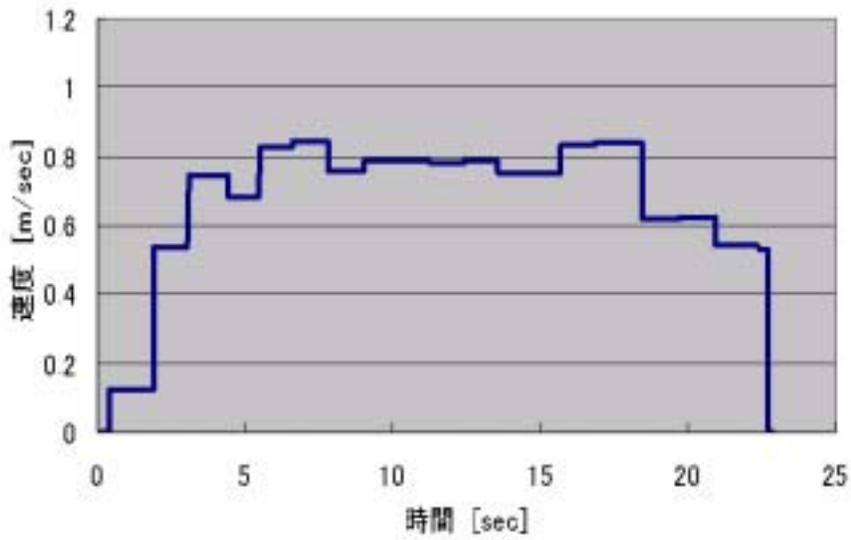


図 5.14: 被験者 2 の速度の時間変化

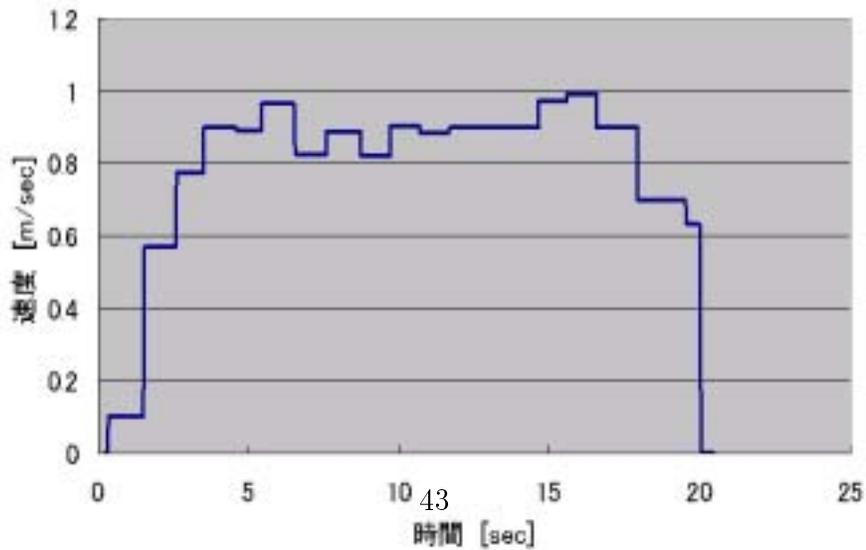


図 5.15: 被験者 3 の速度の時間変化

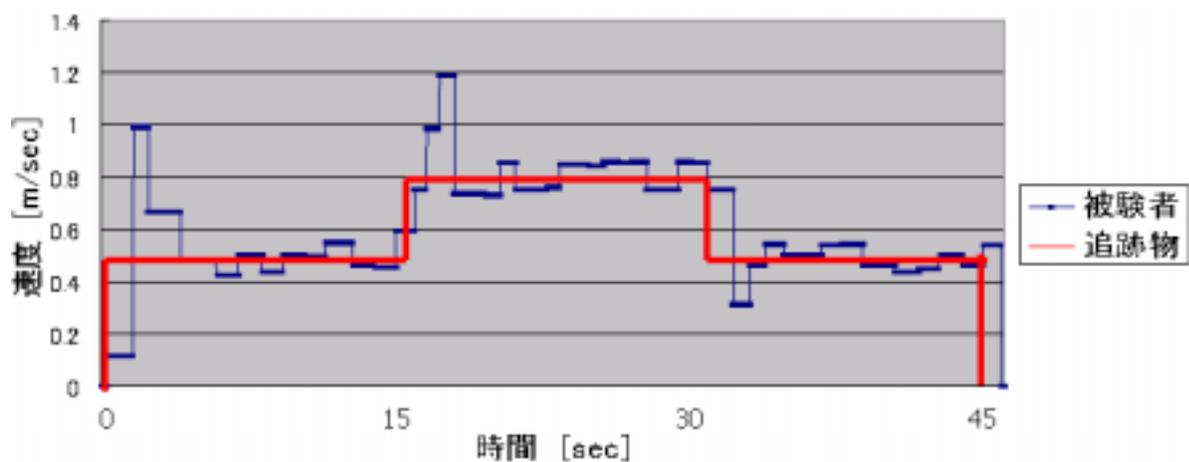


図 5.16: 追跡物と被験者の速度

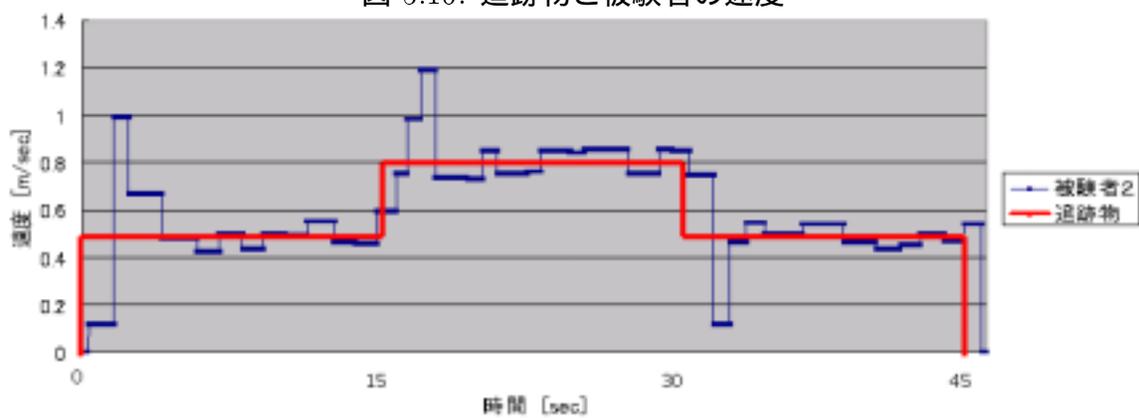


図 5.17: 追跡物と被験者の速度

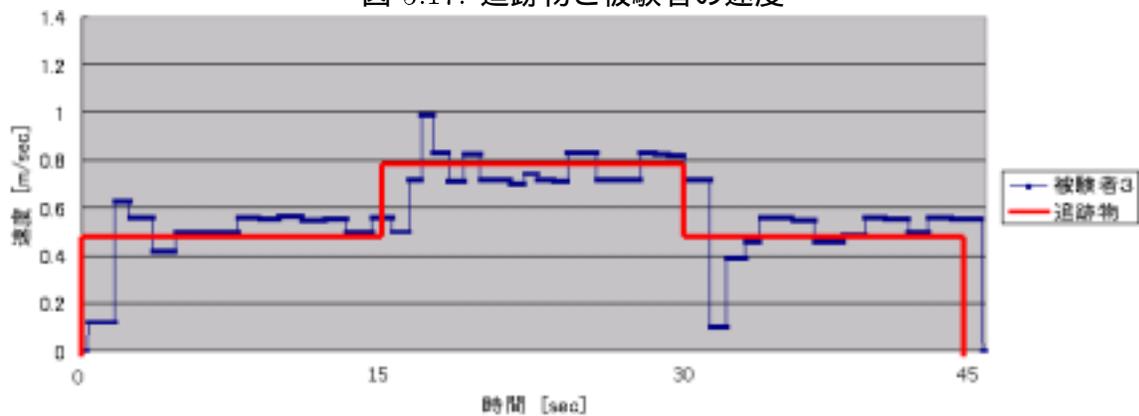


図 5.18: 追跡物と被験者の速度

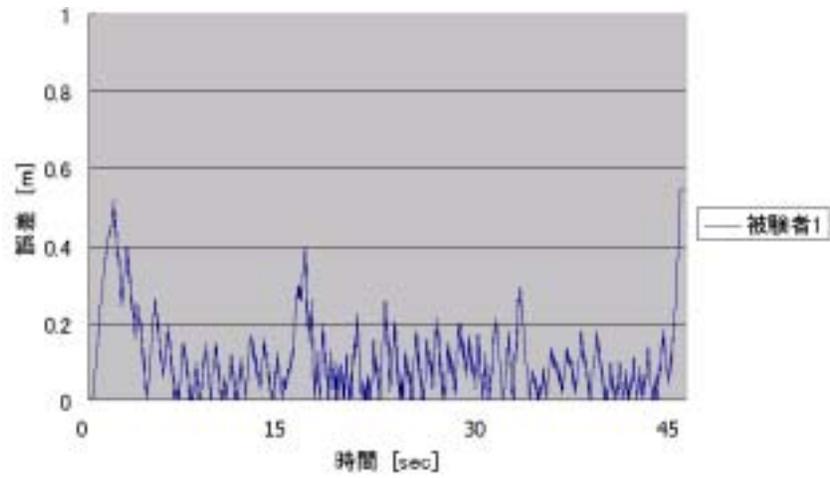


図 5.19: 追跡物と被験者との誤差

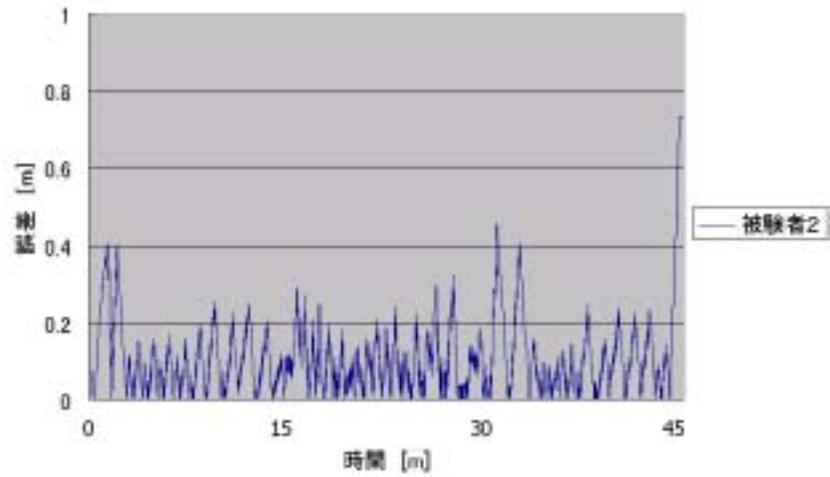


図 5.20: 追跡物と被験者との誤差

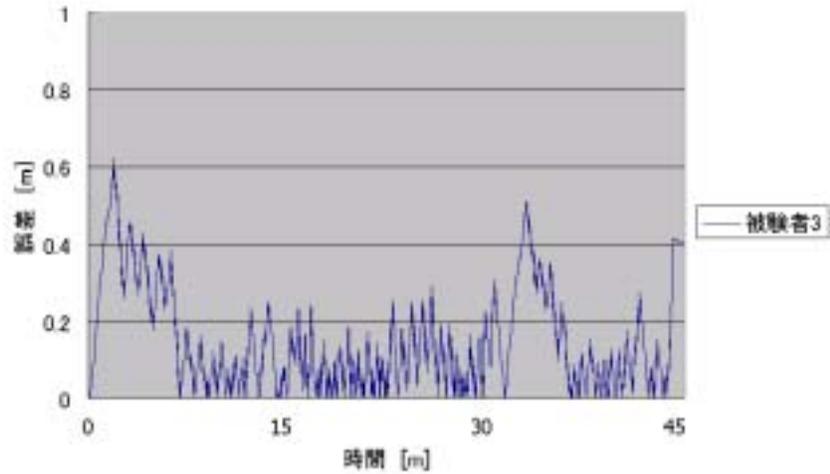


図 5.21: 追跡物と被験者との誤差

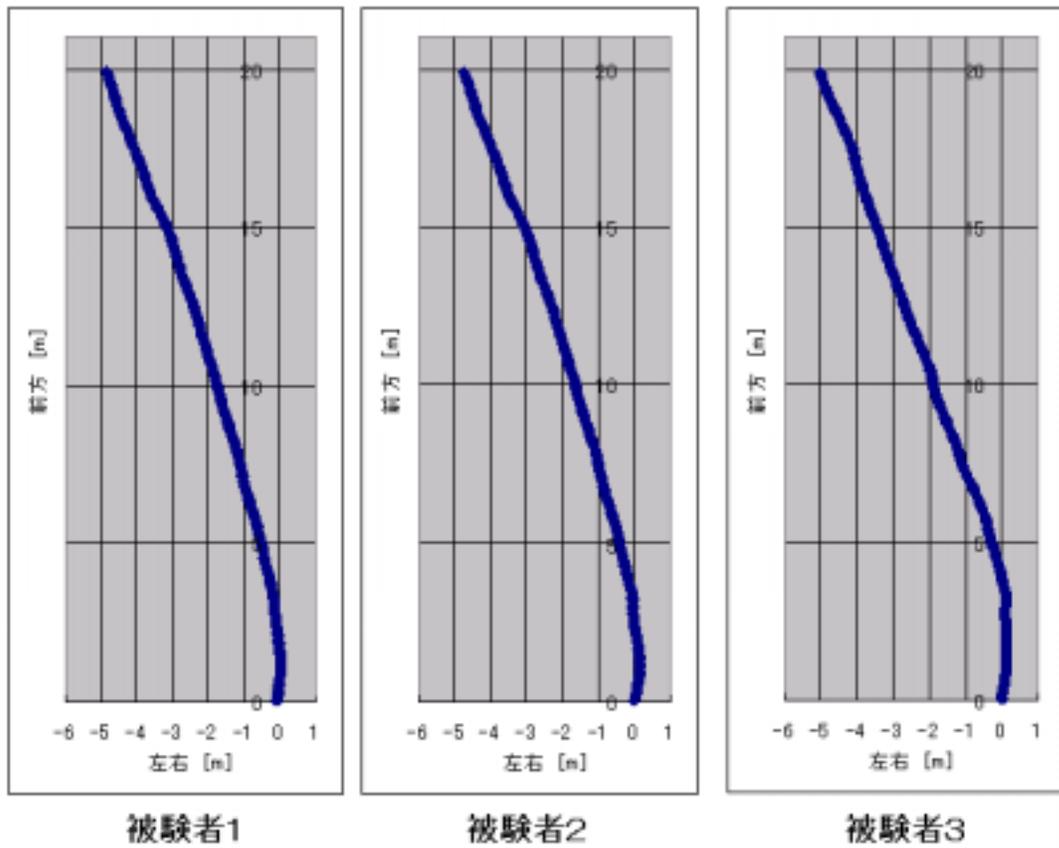


図 5.22: ゴールまでの経路

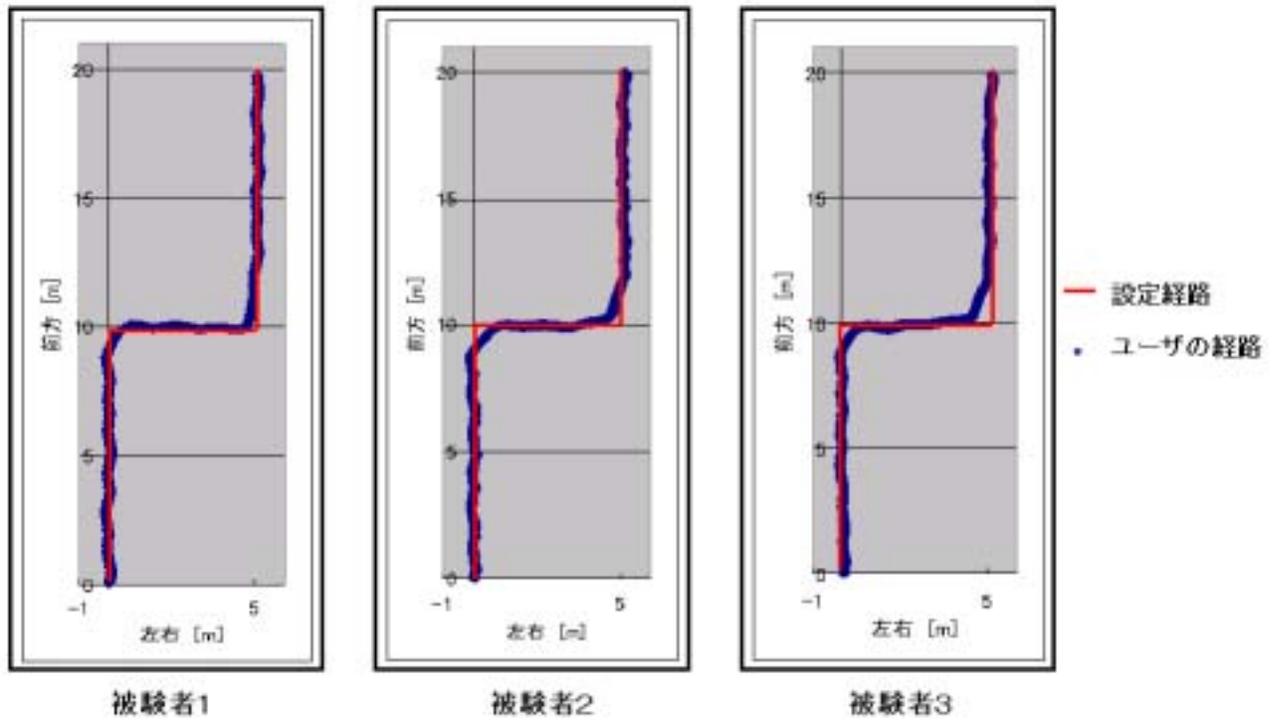


図 5.23: ゴールまでの経路

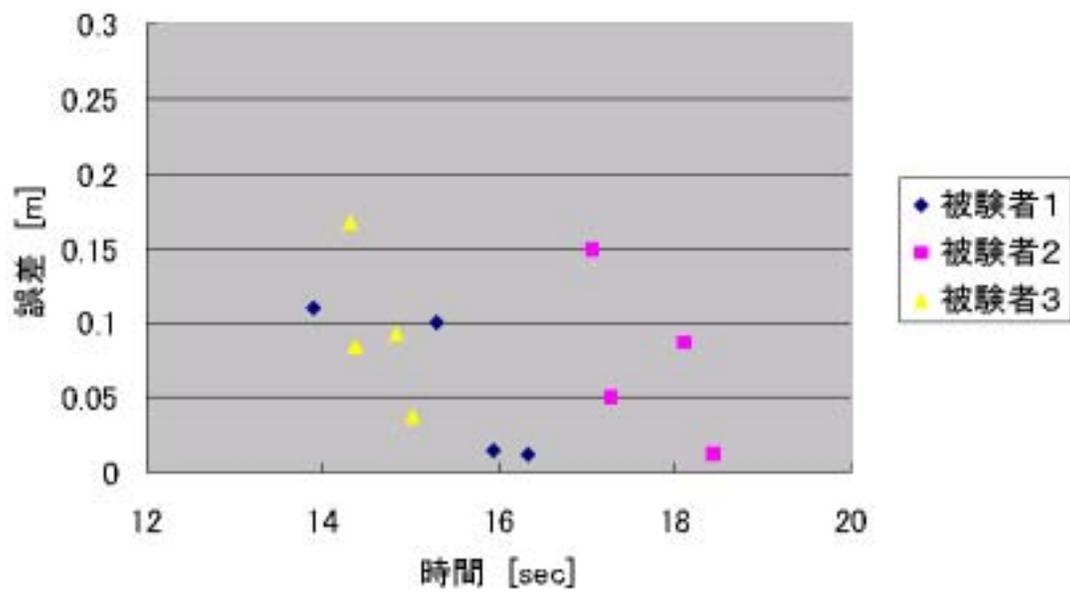


図 5.24: 停止位置誤差



- JOYSTICK_X(+)...回転
[最大 180° /sec ($x = 1$)]
- JOYSTICK_X(-)...回転
[最大 -180° /sec ($x = -1$)]
- JOYSTICK_Y(+)...ウォークスルー
[最大 1m/sec ($y = 1$)]
- JOYSTICK_Y(-)...ウォークスルー
[最大 -1m/sec ($y = -1$)]

図 5.25: 3DWand の構成

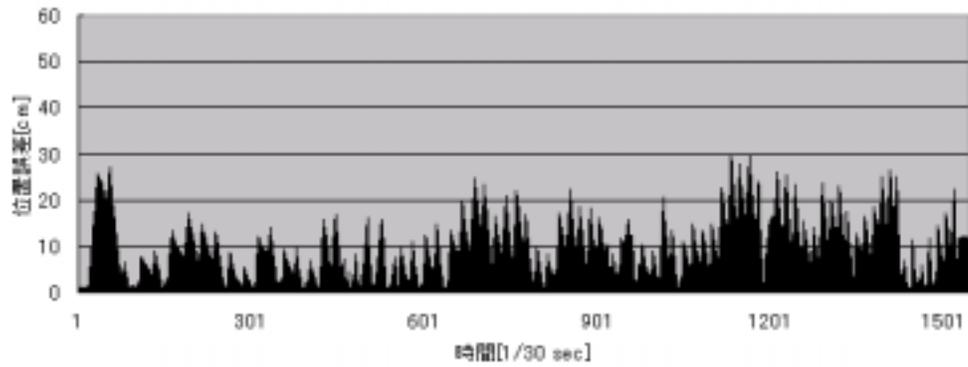


図 5.26: 本システムを用いた被験者 1 の位置誤差

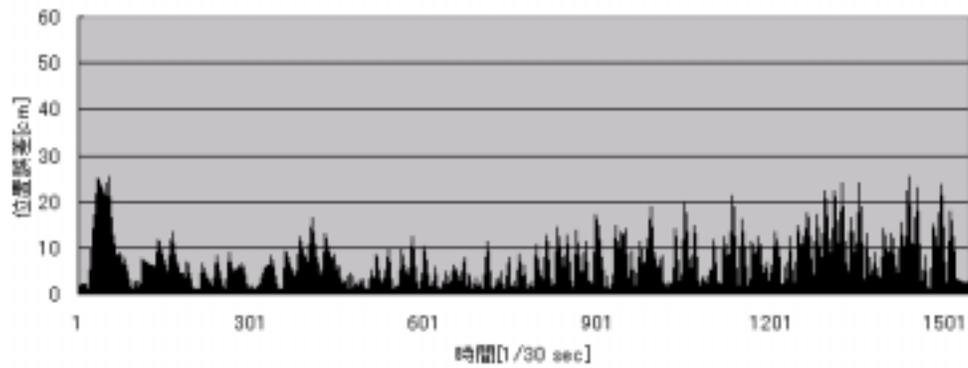


図 5.27: 本システムを用いた被験者 2 の位置誤差

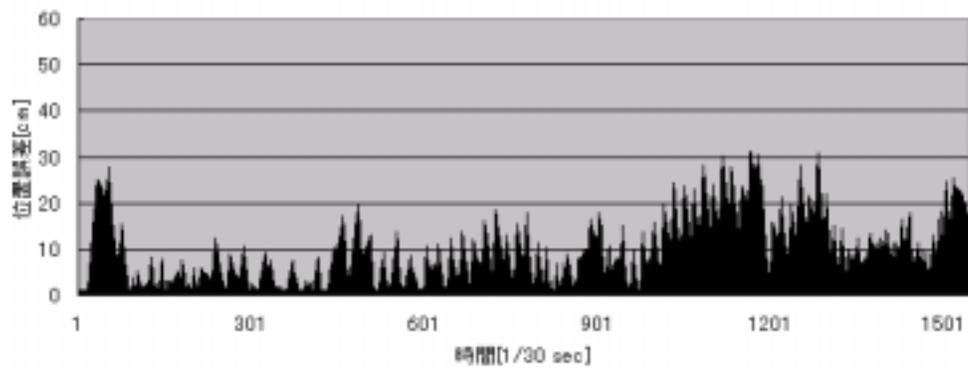


図 5.28: 本システムを用いた被験者 3 の位置誤差

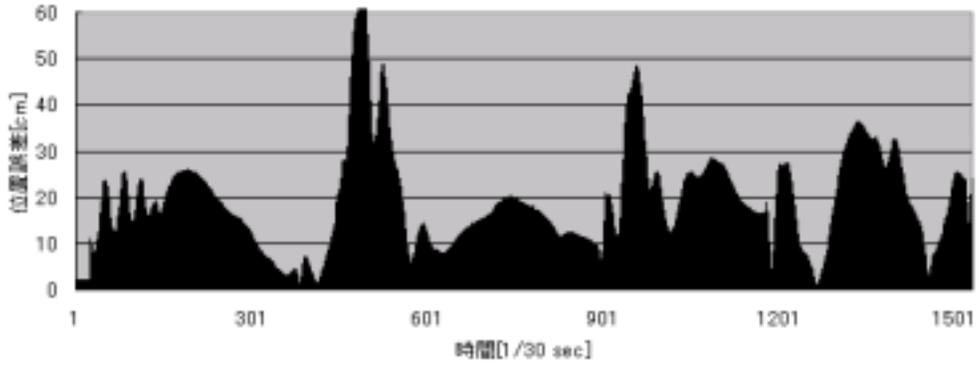


図 5.29: 3DWAND を用いた被験者 1 の位置誤差

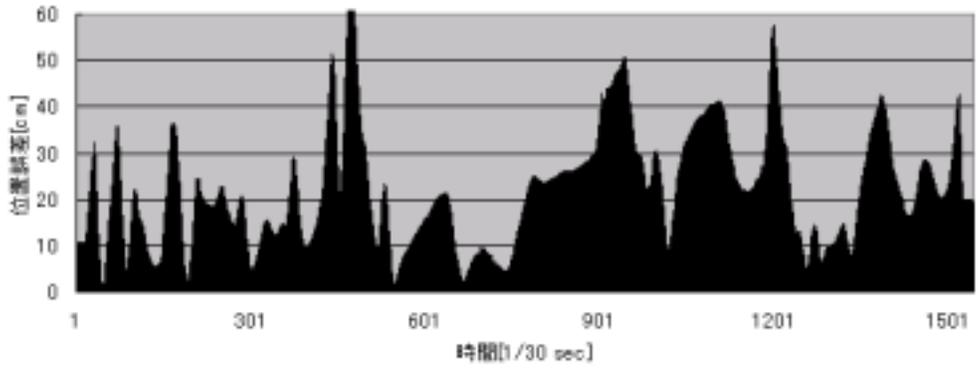


図 5.30: 3DWAND を用いた被験者 2 の位置誤差

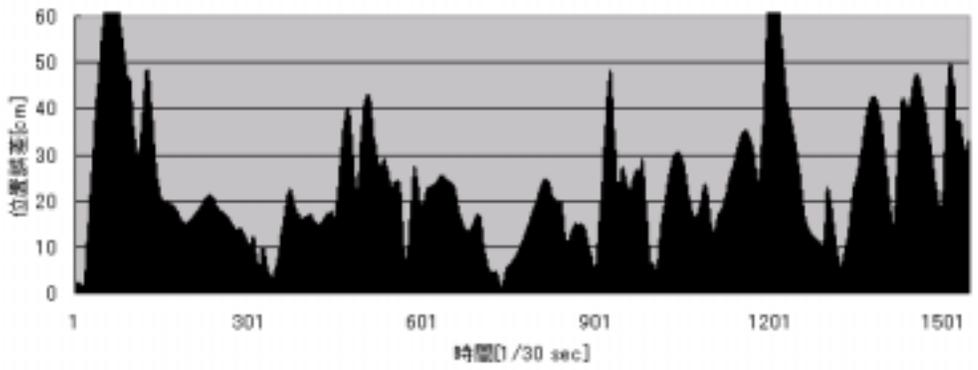


図 5.31: 3DWAND を用いた被験者 3 の位置誤差

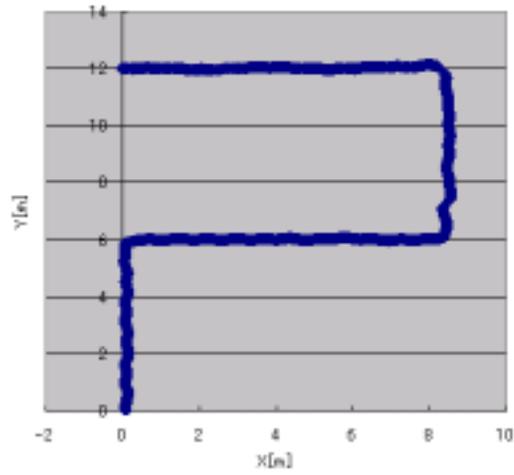


図 5.32: 本システムを用いた被験者 1 の経路

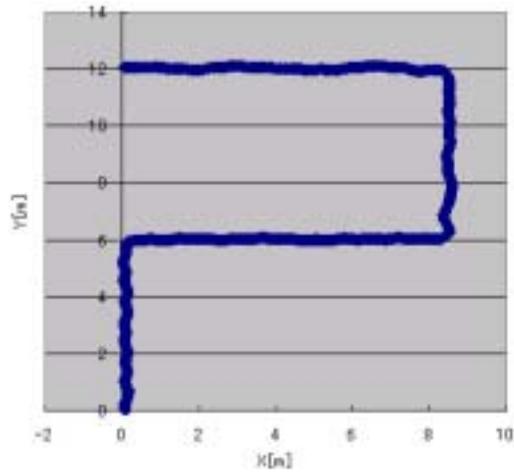


図 5.33: 本システムを用いた被験者 2 の経路

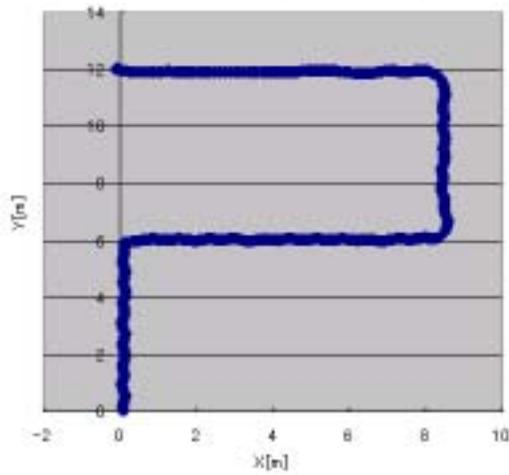


図 5.34: 本システムを用いた被験者 3 の経路

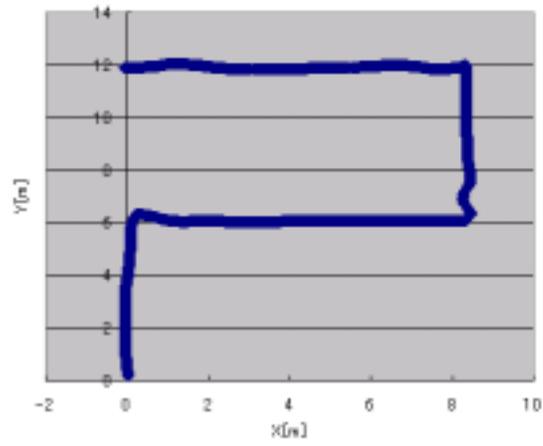


図 5.35: 3DWAND を用いた被験者 1 の経路

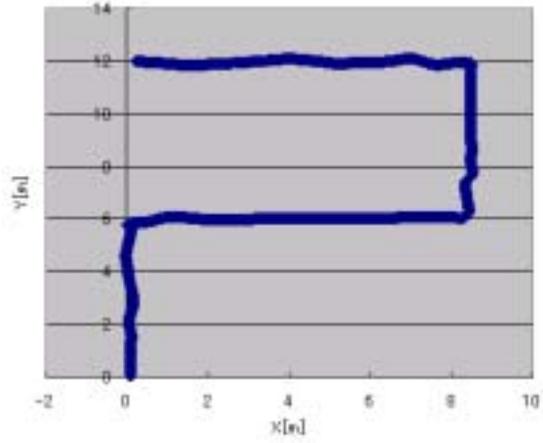


図 5.36: 3DWAND を用いた被験者 2 の経路

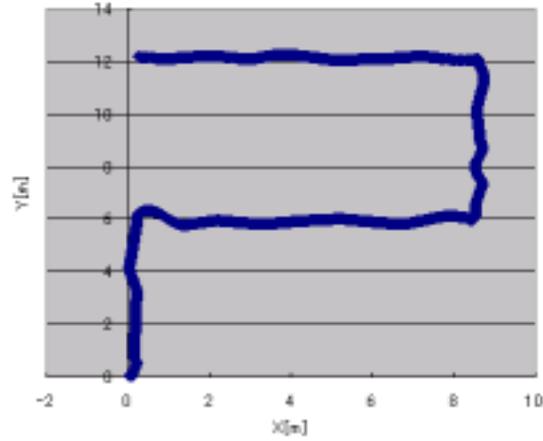


図 5.37: 3DWAND を用いた被験者 3 の経路

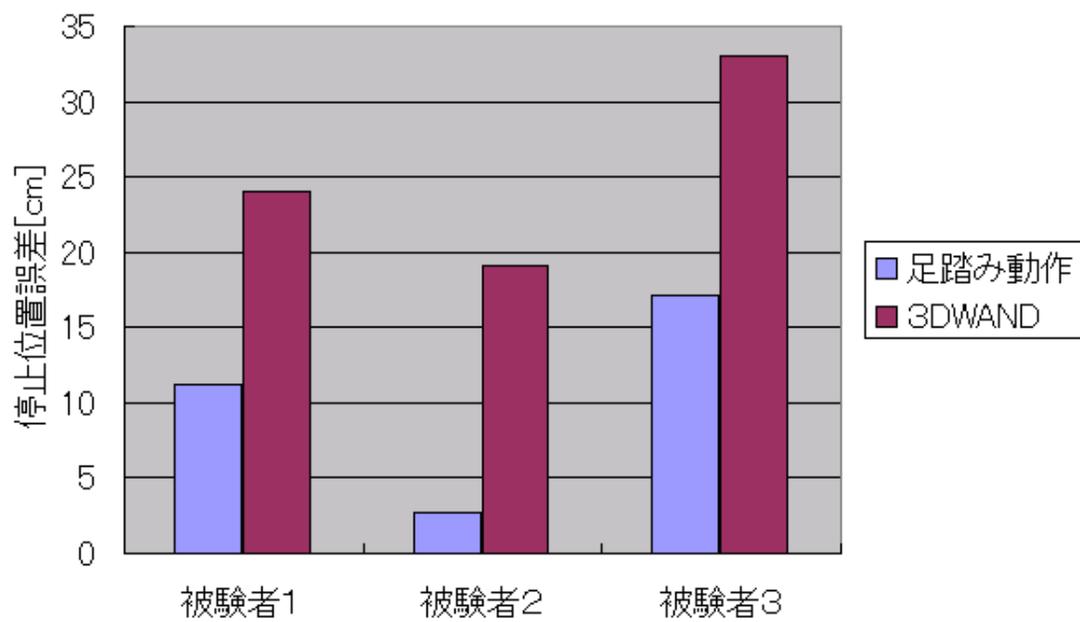


図 5.38: 本システムと 3DWAND の停止位置誤差

第6章 仮想都市空間におけるアバタ構築とコラボレーション

6.1 はじめに

過去に存在した都市や、近未来における都市をコンピュータ上の3次元空間内に構築し、その都市空間内の生活者の営みをシュミレーションすることにより、生活者の視点から都市を評価するなど、都市計画へ応用する事への可能性も期待される。

コンピュータ上に構築した仮想都市空間をよりリアルに演出するための手法の一つに空間内への自分や他人の分身である化身(アバタ)の導入が挙げられる。この仮想都市内にいる人との豊かなコミュニケーションの実現にとって、高い臨場感の提供と共に、日常生活空間の中でシステムを利用することができる。ユーザは自分の視点による映像を見ながら仮想都市空間を歩きまわることができ、現実世界の人間とアバタとの間にコラボレーションが存在させることができる。

そこで、本章では都市生活者の視点から都市を評価するための仕組みを提案する観点から、CAVEに投影した仮想都市空間へモーションキャプチャを利用した人体構造を有するアバタの導入を目的とする。具体的な手法として、構築した仮想都市空間内で3次元のアバタに動作を適用するための仕組みを提案する。そして、モーションキャプチャにより、人物動作認識に必須である身体部位の3次元位置を計測し、アバタをリアルタイムに動作させる。さらに、仮想都市空間内のアバタとコラボレーションさせるために、モーションキャプチャから得た人の動きの3次元位置データをデータベースに蓄積し、それを基に仮想都市空間内でアバタを動作させる。

本章では、仮想空間の人間とのコミュニケーションシステムを構築するために必要となるアバタについて検討を行う。さらに、CAVE内で人物動作を抽出し仮想空間上に反映させる技術、及びこの技術を利用したアプリケーションの実現を述べる。

以降、6.2では、システム構成を述べる。6.3では、人体構造の定義について概説する。6.4では、仮想空間における人体モーションの合成について提案する。6.5では、アバタとのコラボレーションについて述べる。6.6では、考察を述べ、6.7では、まとめを述べる。

6.2 アバタのある仮想都市空間システム構成

今回、本研究で提案、開発した動作共有システムのハードウェア構成図を図 6.1 に示す。ユーザは CAVE 内でモーションキャプチャを装着して動作を行う。モーションキャプチャで計測された 3 次元位置情報は、計算機である Onyx3200 に送られ、この座標値データを基にアバタの動きが制御される。アバタの動作映像はプロジェクターにより CAVE のスクリーンに投影される。

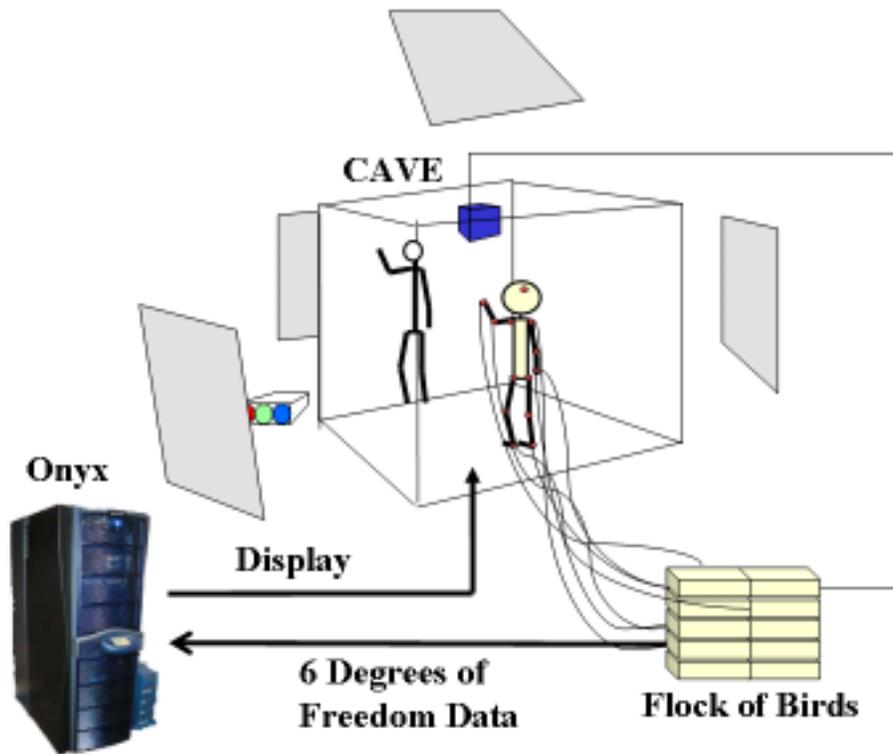


図 6.1: システム構成

6.3 アバタ作成のための人体構造の定義

3 次元コンピュータグラフィックスの普及により、映像アニメーション、ビデオゲーム等で人間の 3 次元モデルを動かす事が行われている。人間の 3 次元モデルの動作を手作業により生成するには時間と労力を要するため、モーションキャプチャを用いてその動作を生成する手法が普及してきた。人間の身体構造をコンピュータで記述するために頻繁に利用されるのがモーションキャプチャのデータである。そこで、本研究では、アバタの動きを制御する方法としては、モーションキャプチャを用いる。ユーザがモーションキャプ

チャを装着して動作を行い、モーションキャプチャによって計算される3次元位置を基にアバタを動かす。

モーションキャプチャは開発メーカーによりいくつか存在するが、現在のところ各々のモーションキャプチャが取り扱う人体構造間に統一した標準はない。本研究では、人間の動作が明確に計測できることを考慮して人体構造を決定した。図6.2に人体構造の定義を示す。本システムにおける人体構造は、15個のジョイント及び14個セグメント(身体の部位)で構成される。人間の体を頭部、首、肩、肘、手首、腰、膝、足首にそうとする15個のジョイントを結合して表している。

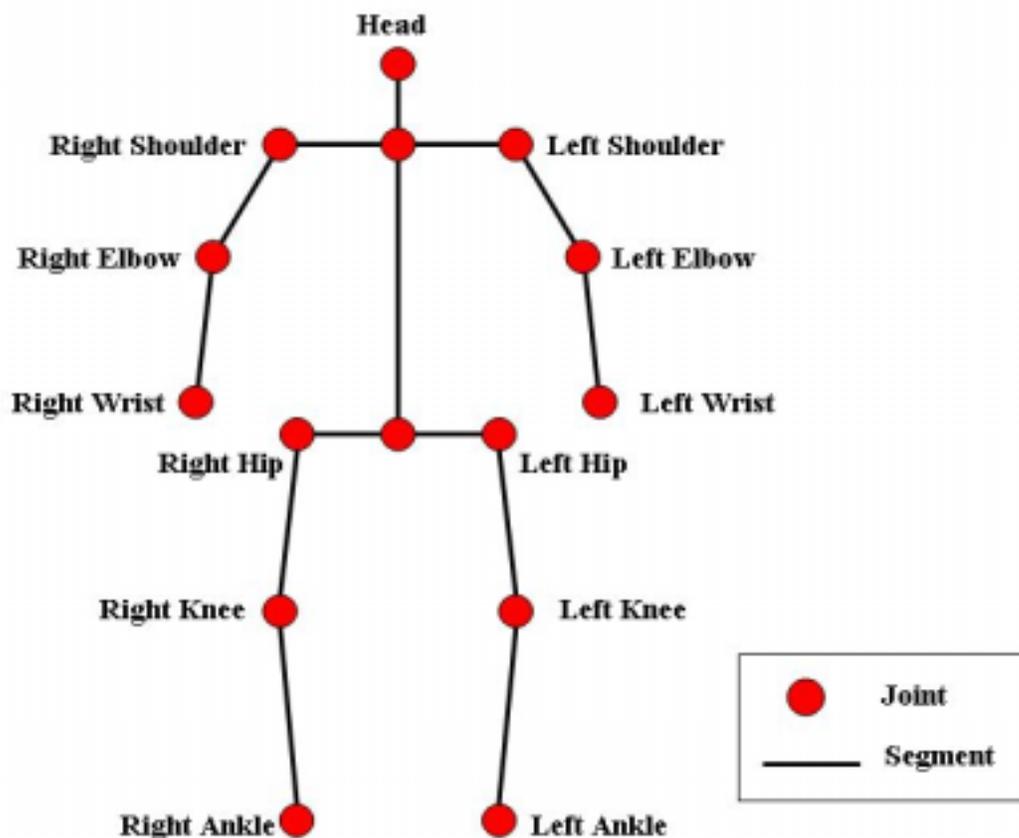


図 6.2: 人体構造の定義

6.4 仮想空間における人体モーションの合成

人間の動きをモーションキャプチャを用いて計測し、アバタに動きを合成することを提案する。これらにより人間らしい動きの基礎を与えることができる。人間と同じ動作をさせるアバタを表現するには、表示されるアバタの大きさや表示位置を制御して、ユーザの

身体の大きさ、3次元位置に合わせる必要がある。そこで、本研究では、モーションキャプチャを身につけた人の3次元位置データと仮想都市内のアバタの3次元位置データを一致させることによりこれを解決した。

まず、人に前節で述べた人体構造のジョイント部位と同じ位置にセンサーを取り付ける。人に身につけたセンサーの3次元位置データをそのまま仮想空間の世界座標に変換し、その3次元位置でアバタのジョイントを決定する。アバタのセグメントは人体構造となるようにジョイント間を結ぶことで表現している。アバタは、OpenGLのプリミティブを使用し、簡易的なオブジェクトモデルで仮想環境内に再現した。図6.3に各ジョイントをサンプリングポイントにしセンサーをとりつた様子とそれに対応したアバタの様子を示す。このように、実世界で動く人の3次元位置情報をそのまま仮想空間の世界座標に変換しているため、CAVEに投影されるアバタは、ユーザの身体の大きさと同じ大きさで表現される。さらに、リアルタイムに3次元位置データを変換しアバタのジョイントを決定しているため、実世界の人間の動きに合わせてアバタを動作させることができる。

そこで、アバタの動作検証のため、ユーザに様々な動作をしてもらいアバタを動作させることを試みた。図6.4、図6.5、図6.6にユーザがアバタに様々な動きを動作させている様子を示す。図6.4では、ボールを投げている動作、図6.5はボールを蹴っている動作、図6.6は、殴っている動作をしている様子である。検証の結果、アバタはユーザと同じ動きをしていることが確認された。

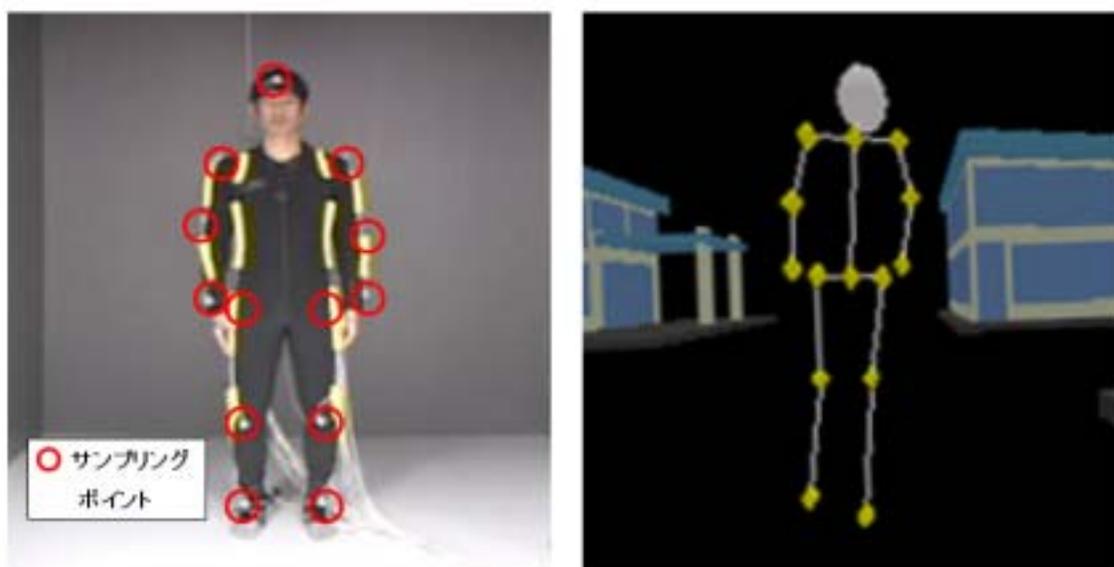


図 6.3: センサーを身につけたユーザと仮想都市空間に導入したアバタ

6.5 仮想都市空間でのアバタとのコラボレーション

近年、仮想都市空間内で生活者の営みをシュミレーションすることにより、生活者の視点から都市を評価するなど、都市計画へ応用する事への可能性も期待されている。また、人は日常のコミュニケーションにおいて、お互いの意図や感情をより深く理解するには、単に相手から発せられる言葉やだけでなく、態度や動作といった情報が重要になる。そこで、人と同じ動作をするアバタを仮想都市内へ導入することによって、ユーザは自分の視点による映像を見ながら仮想都市空間を歩きまわり、仮想都市内のアバタと豊かなコミュニケーションが実現することができる。

そこで、本研究では仮想都市空間内のアバタとコラボレーションさせるために、モーションキャプチャから得た人の動きの3次元位置データをデータベース化し、それを基に仮想都市空間内でアバタを動作させた。あらかじめ人間の動作を3次元位置情報を基に計測してOnxyのハードウェア上でデータベース化する。システムの起動後にデータベースから3次元位置データを読み込み、アバタの3次元位置データとして入力することでアバタを動作させた。

このシステムを利用することによって、ユーザは、仮想都市空間内のアバタと豊かなコミュニケーションが実現することができる。図 6.7 にユーザとアバタとが握手してコミュニケーションをしている様子を示す。このように、現実世界の人間と仮想都市内の人間との間にコラボレーションを実現した。

6.6 考察

アバタを導入した本システムの応用を考察する。CAVEでは立体視が可能であるので、等身大のアバタを立体映像として表示することで、人間がそのアバタに重なることが可能となる。アバタに重なることによりアバタの身ぶりと自己の身ぶりを見ることができる。これはスポーツフォームやダンス等の学習に活用することができる。例えば、スポーツフォームを覚える場合、アバタの動きが模範動作となっていれば、ユーザは常にアバタに重なるうとして身ぶりを行えばよい。

また、これまで様々なスポーツ動作を客観的に参照するには、ビデオなどの映像情報を利用して来た。映像情報に対し、3次元デジタル化されたモーションデータは様々な方向からの参照であるため、スポーツ動作を分析するアプリケーションとしても有用である。

本論文での提案システムをネットワークを返して共有し、自分の動作を遠隔地のシステム上でのアバタとすることができれば、離れた場所にいる者同士がお互いの動作を共有することが可能になる。しかも、何か3次元の仮想物体に対して協調作業したりすることが可能となる。協調作業機能を用いれば、仮想のボールを使ってキャッチボールを行うことも可能であろう。

6.7 まとめ

本章では、仮想都市内でのアバタを導入することの必要性を述べ、アバタについて検討を行った。アバタの人体構造を定義し、3次元CG技術を用いてアバタのモデルを構築した。このアバタをモーションキャプチャの3次元位置データを基にリアルタイムに動作させるための提案およびシステムを構築した。さらに、仮想的に構築した仮想都市空間にアバタを導入することで、アバタと人間との間でコミュニケーションを実現した。最後に、本システムを利用したアプリケーションの実現に関して考察した。

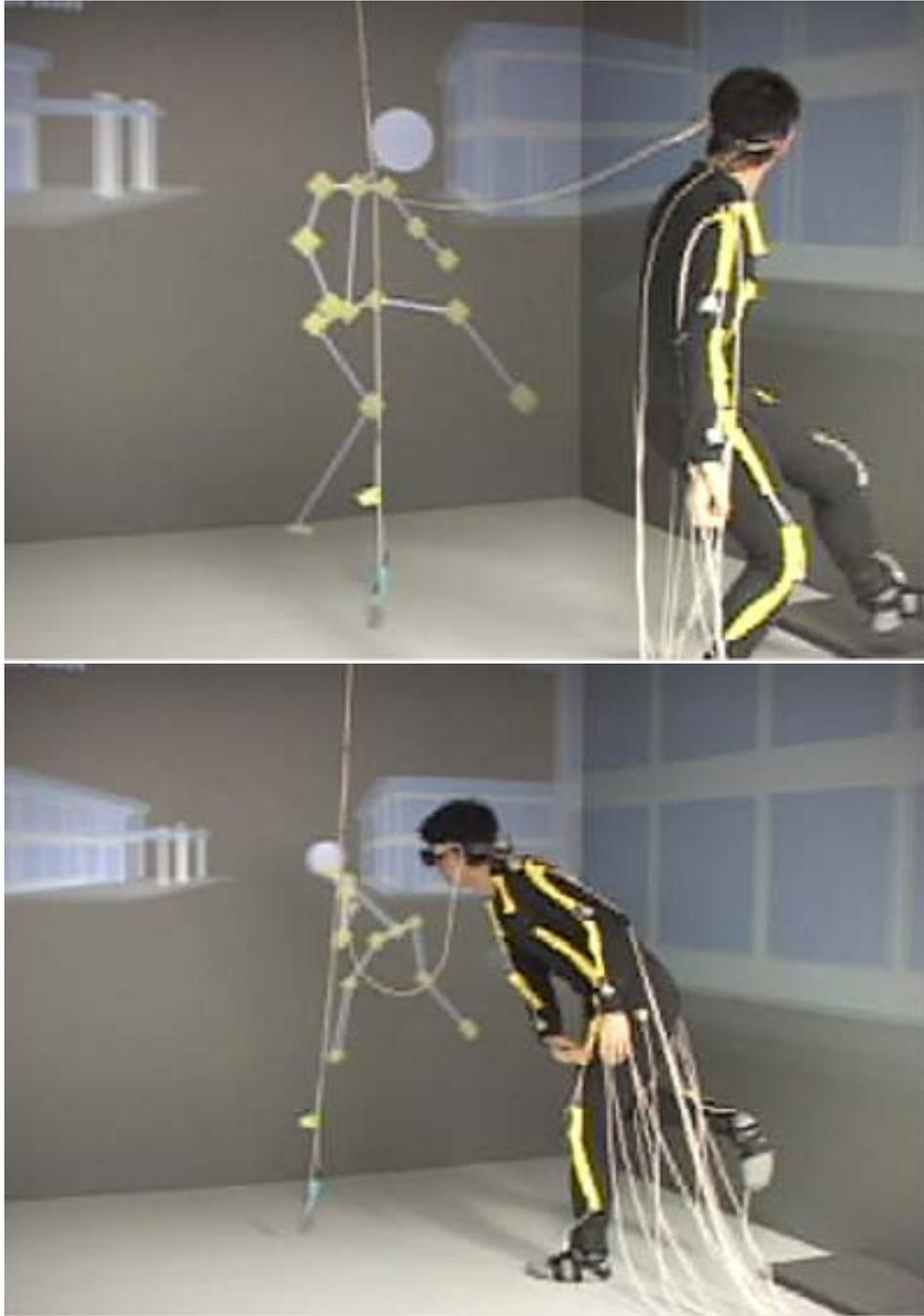


図 6.4: ボールを投げる動作

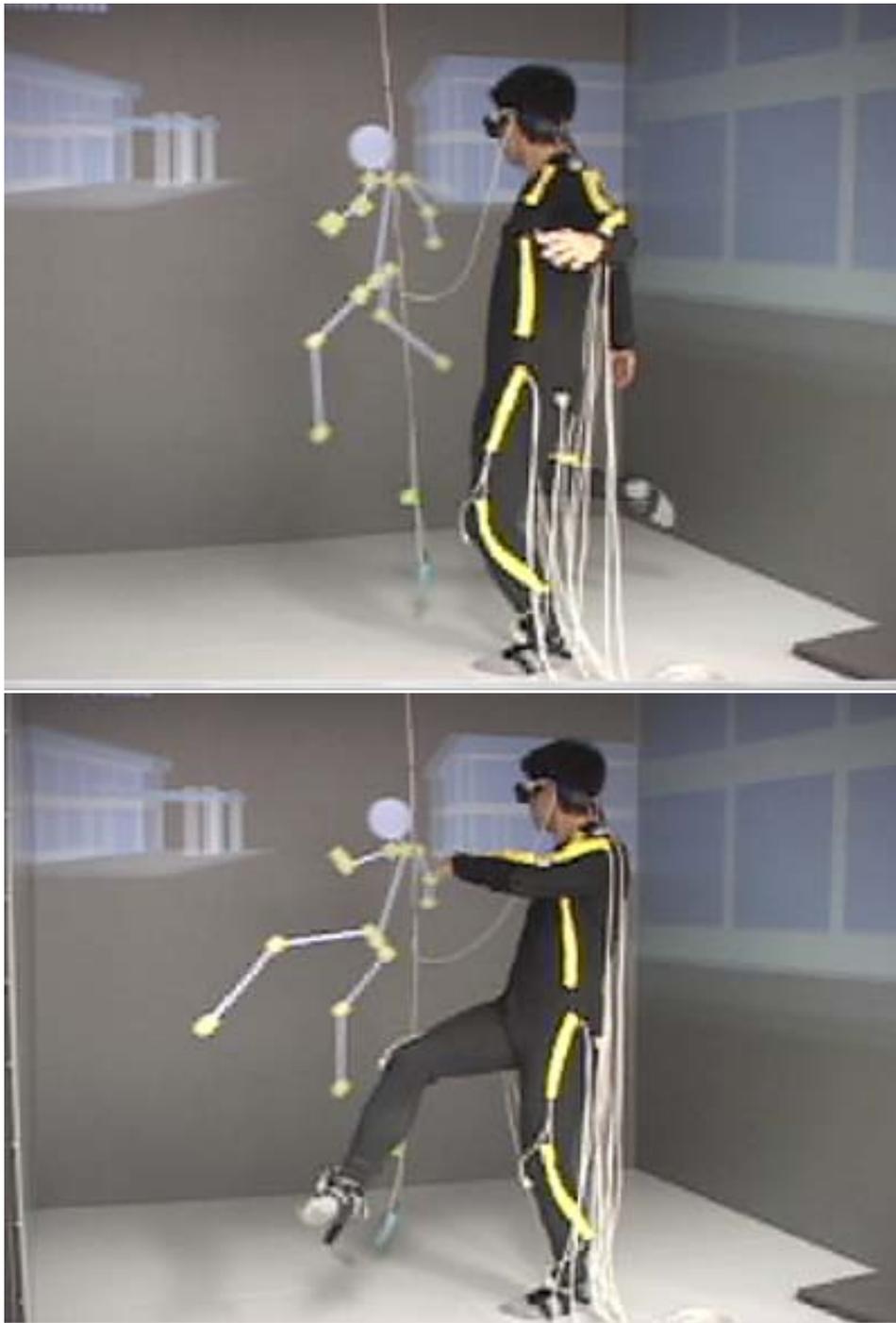


図 6.5: ボールを蹴る動作

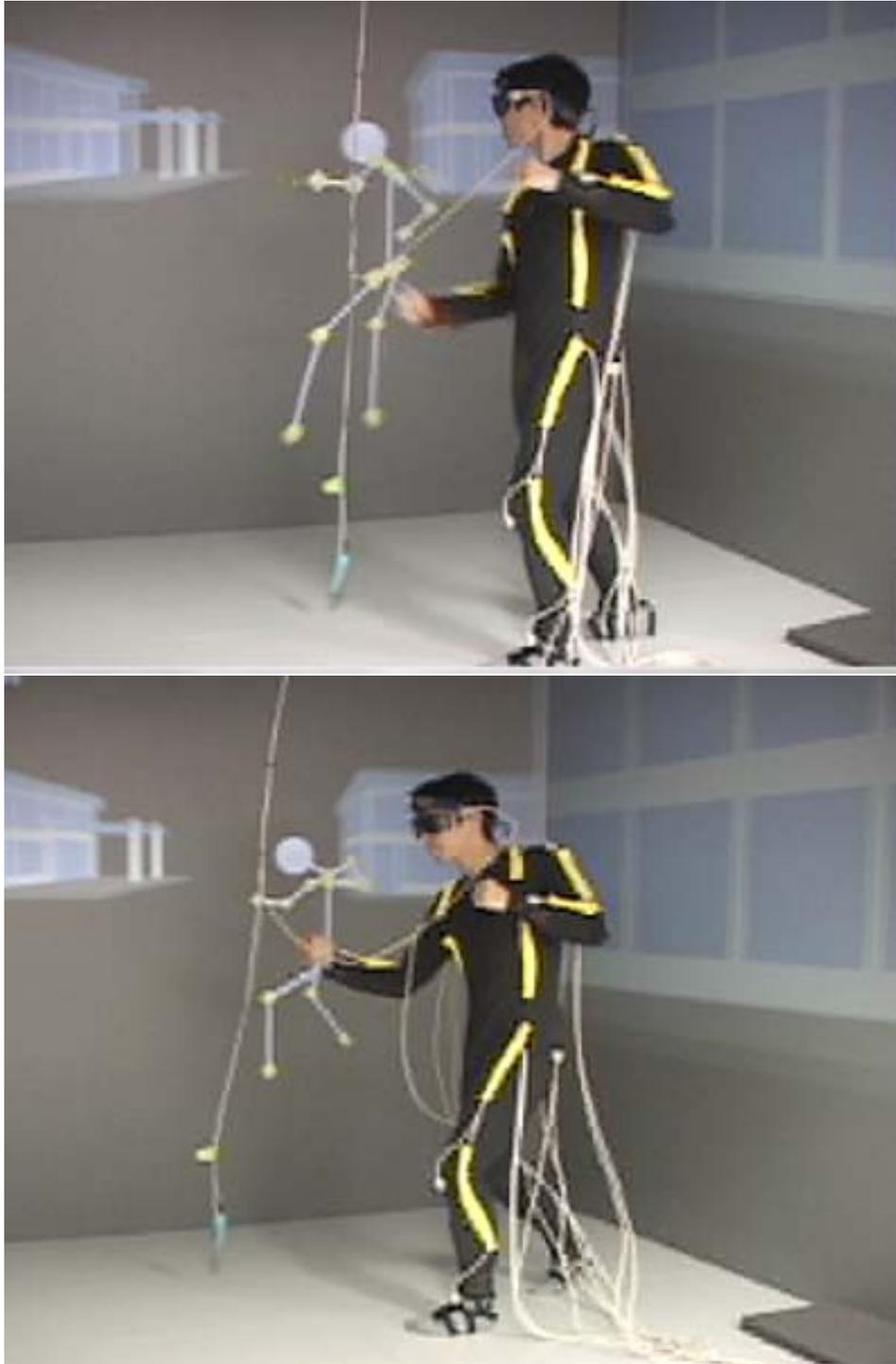


図 6.6: 殴る動作

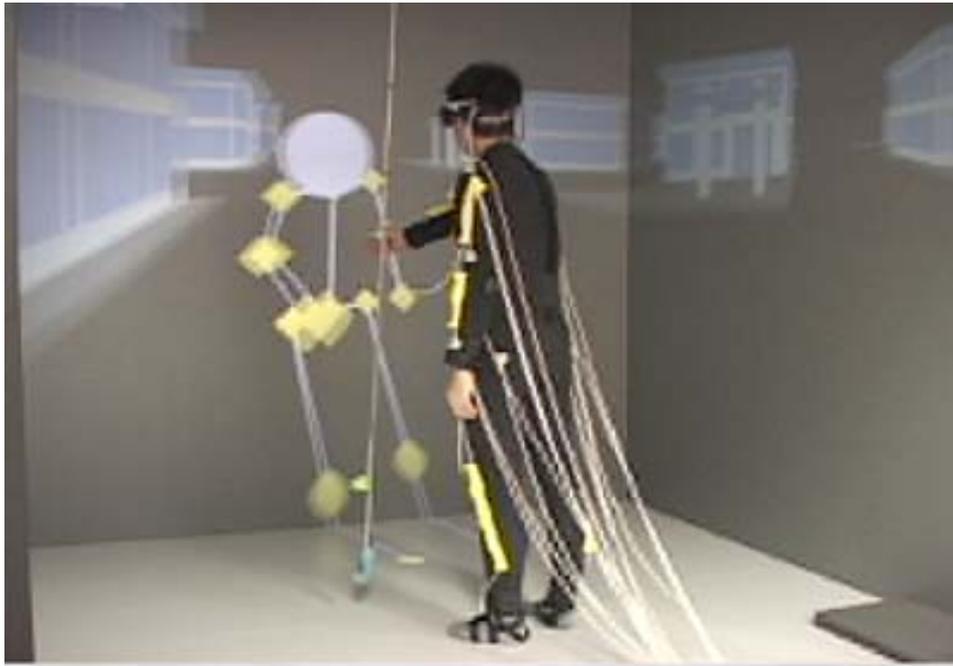


図 6.7: アバタと握手している様子

第7章 まとめ

7.1 まとめ

都市や住宅などを設計する場合、等身大の仮想デジタルシティを構築することで、あらかじめ安全性や快適性の評価を行うことができる。臨場感のある等身大の仮想都市環境により、その都市を仮想体験することで、都市の問題点を体感することができ、都市の評価、設計に役立つと期待されている。このような仮想環境では、実世界で人間が見たり動作したりすることと同じような体験ができることが要求されており、実世界の人間の動作と同じようにインタラクションすることが要求されている。このように、インタラクションインタフェースも実世界と同じような動作でインタラクションする必要がある。

本論文では、モーションキャプチャを用いることによって人間の動作の3次元データを正確に取得し、人体の動きに応じて仮想空間と直感的にインタラクションするインタフェースを構築した。

第2章では、従来研究のインタラクションインタフェースを説明し、仮想都市内を自然な動作でインタラクションをするのに不十分であること明らかにした。

第3章では、仮想空間の移動手段として、足踏み移動による仮想空間内のインタフェースを構築した。実際の歩行速度とピッチの間に比例関係が見られたため、本研究ではピッチから移動速度を推定する手法を採り、これを基に足踏み歩行により仮想空間を歩行するインタフェースを提案した。

第4章では、仮想物体操作として、掴む動作による仮想物体操作インタフェースを構築した。仮想物体操作するには、仮想物体を指し示すための1点の座標と選択したとうアクションが必要である。本研究では、1点の座標を手先とし、選択したというアクションを人が実世界で行う動作に近い手を握る動作として採用した。

第5章では、これらの提案した手法を用いて、仮想都市インタラクションシステムを構築した。実現した仮想都市インタラクションシステムの中で、従来のインタラクションインタフェースである3DWANDと比較実験を行い、本インタフェースの有効性を確認した。

第6章では、仮想都市空間内のアバタとコラボレーションさせるために、モーションキャプチャから得た人の動きの3次元位置データをデータベース化し、それを基に仮想都市空間内でアバタを動作させた。

謝辞

本研究を行うに当たり御指導御鞭撻を戴いた北陸先端科学技術大学院大学堀口 進教授に深く感謝致します。

またサブテーマで御指導を戴いた敷田幹文助教授に厚く御礼申し上げます。

林 亮子助手には、貴重なコメントを戴き深く感謝申し上げます。

多くの有益なご指摘を戴きました福土助手に深く感謝致します。

基本仕様や設計等様々な面で支援して下さった堀井 洋助手に深く感謝申し上げます。

最後に、日頃よりお世話になった堀口研究室の皆様に厚く御礼申し上げます。

参考文献

- [1] 久木元伸如, 手の振りをインタフェースとした仮想空間の移動, ヒューマンインタフェース学会, Vol.1, No. 2, pp. 11-14, 1999.
- [2] 小林稔, 志和新一, 北川愛子, 市川忠嗣, 一之瀬進: 体重移動によるサイバースペース内移動インタフェース, 電子情報通信学会論文誌, 98/5 Vol.J8I-D-II, No.5.
- [3] 中川千鶴, 大須賀美恵子, VE 酔い研究および関連分野における研究の現状, 日本 VR 学会論文誌, Vol3, No2, pp.31-38, 1998.
- [4] C.Schmandt, Spatial input/output correspondence in a stereoscopic computer graphic work station, Computer Graphics, vol.17, no.3, pp.253-261, ACM July 1983.
- [5] G.W.Fitzmaurice, S.ZZhai, and M.H.Chignell, Virtual Reality for Palmtop Computers, ACM Transaction on Enformation Systems, Vol.11, no.3, pp.197-218, July 1993.
- [6] C.Lee, J.Lee, C.Park, and D.Kim, Real-time gesture recongition for the control of avatar, 日本バーチャルリアリティ学会第 2 回大会論文集, pp.242-245, Sept 1988.
- [7] H.Iwata and K.Matsuda, Haptic walkthrough simulator: Its design and application to studies on cognitive map, Proc. of the 2nd International Conference on Artificial Reality and Tele-existence, pp.185-192, 1992.
- [8] H.Iwata and T.Fujii, Virtual perambulator: A novel interface device for locomotion in virtual environment, Proc. of IEEE 1996 Virtual Reality Annual International Symposium, pp.60-65, March 1996.
- [9] M.Slater, M.Usho, and A.Steed, Taking Steps: The influence of a walking techique on presence in virtual reality, ACM Transactions on Computer Human Interaction, vol.2, no.3, pp.201-219, Sept 1995.
- [10] F.P.Brooks, Jr., J.Airey, J.Alspaugh, A.Bell, R.Brown, C.Hill, U.Nimscheck, P.Rheingans, J.Rohlf, D.Smith, D.Turner, A.Varshney, Y.Wand, H.Weber, and

- X.Yuan, Six Generations of Building Walkthrough: Final Technical Report to the National Science Foundation, Department of Computer Science, University of North Carolina, Chapel Hill, June 1992.
- [11] 岩田洋夫, 吉田陽子, 無限平面を用いた仮想歩行装置 日本バーチャルリアリティ学会 第2回大会論文集, pp.254-257, Sept 1997.
- [12] F.M.Fairchild, L.B.Hai, J.Loo, N.Hern, and L.Serra, The heaven and earth virtual reality: Designing applications for novice users, Proc. of IEEE 1993 Virtual Reality Annual International Symposium, pp.47-53, Sept 1993.
- [13] 田中幹大, 根本啓次, VRを利用したスキートレーニングシステムの実現, テレビ学技報, Vol.19, no21, pp.13-18, March 1995.
- [14] <http://www.scl.ameslab.gov/HINT>
- [15] Philip R. Cohen, Michael Johnston, David McGee, Sharon Oviatt, Jay Pittman, Ira Smith, Liang Chen and Josh Clow, Multimodal interaction for Distributed Applications, In Proceeding of the 1986 Workshop in Interactive 3D Graphics,
- [16] Brogan, D. C., Metoyer, R. A., and Hodgins, J. K, Dynamically Simulated Characters in Virtual Environments, IEEE Computer Graphics and Applications, pp. 58-69. September October 1998, Volume 15 Number 5.
- [17] Joseph J. LaViola, Jr., Daniel Acevedo Feliz, Daniel F. Keefe, Robert C. Zeleznik, Hands-free multi-scale navigation in virtual environments, symposium on Interactive 3D graphics, Pages: 9 - 15, Proceedings of the 2001.
- [18] 平田幸広, 江島由規, 石井雅博, 佐藤誠: ウォークスルーのための足踏み動作の解析, 社団法人 電子情報通信学会 信学技報, MVE2000-36 (2000-06).
- [19] 野間春生, 宮里勉, 中津良平: 能動的歩行動作に対応した歩行感覚提示装置の開発, 日本バーチャルリアリティ学会論文誌, Vol.4, No.2, 1999.
- [20] 篠田康宏, ラルーシ ブギラ, 長谷川晶一, 小池康春, 佐藤誠, 仮想都市環境のためのマルチモーダルインタフェースの開発, 社団法人 電子情報通信学会 信学技報, MVE2000-116 (2001-03).
- [21] 中村, 斎藤, 基礎運動学, 医歯薬出版 (1976).