

NAIST-IS-DT9661003

博士論文

3次元ユーザインタフェースと拡張現実感を用いた
情報ブラウジング手法に関する研究

大隈 隆史

1999年3月24日

奈良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

本論文は奈良先端科学技術大学院大学情報科学研究科に
博士(工学) 授与の要件として提出した博士論文である。

大隈 隆史

審査委員： 横矢 直和 教授
千原 國宏 教授
佐藤 宏介 助教授

3 次元ユーザインタフェースと拡張現実感を用いた 情報ブラウジング手法に関する研究*

大隈 隆史

内容梗概

計算機ネットワーク環境の爆発的な普及は大規模な情報を利用する機会を増加させ、情報のブラウジング手法への注目を集めた。情報のブラウジング時には対象となる情報の全体構造と注目部分の詳細な内容が同時に把握できることが望ましい。しかし、一般的な計算機環境ではこれらを同時に把握する環境は提供されていない。そこで、本論文では情報の全体構造と注目部分の詳細な内容の同時把握を可能にする情報作業空間の構築を目的とする。まず1章で情報作業空間の発展について述べ、本研究の目的と意義を整理する。次に2章では、情報構造を効果的にブラウジングするための情報視覚化手法について検討し、順序付き階層構造の3次元視覚化手法を提案する。提案手法の有効性はブラウジングを想定した評価実験により確認された。3章では情報作業空間全体と注目部分の詳細を同時に提示するために、CRTと透過型Head Mounted Display(HMD)を併用する拡張現実感による情報提示手法を提案する。提案手法の有効性は評価実験により確認された。4章では3章の手法で用いる拡張現実環境における位置ずれ問題をコンピュータビジョンの手法を適用して解決し、ビデオシースルー型拡張現実環境提示システムを試作してその有効性を確認した。最後に5章で全体を総括する。

キーワード

3次元ユーザインタフェース、情報視覚化、拡張現実感、実時間カメラパラメータ推定

*奈良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 博士論文, NAIST-IS-DT9661003, 1999年3月24日.

A Method for Browsing Information Using a 3D User Interface and Augmented Reality*

Takashi Okuma

Abstract

This paper describes methods to construct an environment for browsing information effectively. Recently, methods for browsing information are attracted much attention because of increase in the diffusion of the computer network. When a user browses information, it is helpful for the user to show both of the whole structure of the information and the detail of a important part of the information. Chapter 1 gives traditional methods for browsing information as backgrounds of this research, and describes the goal of this research. In chapter 2, Spiral Trees are proposed as a 3-D visualization method of hierarchical ordered information. Chapter 3 describes the method to enlarge information work area using an augmented reality. This method gives users a augmented environment that has high-resolution and large work area. Chapter 4 discuss a method to calibrate camera parameters exactly in real-time. This method can reduce the registration errors of the method to enlarge work area proposed in chapter 3. A prototype system of video see-through augmented environment is implemented to confirm the effectiveness of the method. Chapter 5 gives the conclusion.

Keywords:

information visualization, 3-D user interface, augmented reality, real-time camera calibration

*Doctor's Thesis, Department of Information Systems, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DT9661003, March 24, 1999.

目次

1. 序論	1
1.1 情報作業空間と情報のブラウジング	1
1.2 本研究の目的	3
1.2.1 順序付き階層構造の3次元情報視覚化	3
1.2.2 拡張現実感技術による情報作業空間の拡張	4
1.2.3 実環境画像からの実時間カメラ位置推定手法	6
1.3 本論文の構成	7
2. 順序付き階層構造情報の3次元視覚化	8
2.1 諸言	8
2.2 Cone Tree：階層構造の3次元視覚化手法	10
2.3 Spiral Tree：順序付き階層構造の3次元視覚化手法	13
2.3.1 順序の利用	13
2.3.2 ノードの配置	14
2.4 実験1：Cone TreeとSpiral Treeの比較実験	15
2.4.1 目的と設定	15
2.4.2 結果	18
2.4.3 考察	19
2.5 実験2：階層内順序の利用に関する評価実験	20
2.5.1 目的と設定	20
2.5.2 結果	21
2.5.3 考察	23
2.6 考察	23
2.7 結言	26
3. 拡張現実感による3次元情報作業空間の拡大	28
3.1 諸言	28
3.2 情報作業空間提示領域の拡大手法	29

3.3	実験 3：目標探索タスクによる評価実験	33
3.3.1	目的と設定	33
3.3.2	結果	34
3.3.3	考察	35
3.4	実装上の問題点	36
3.5	実験 4：仮想物体移動タスクによる評価実験	38
3.5.1	目的と設定	38
3.5.2	結果と考察	41
3.6	考察	46
3.7	結言	47
4.	画像からの視点位置推定による実環境と仮想環境の幾何的整合	48
4.1	諸言	48
4.2	画像からのカメラパラメータ推定	49
4.3	マーカ追跡による実時間位置合わせ	51
4.3.1	処理の概要	51
4.3.2	マーカの追跡	52
4.3.3	カメラパラメータの計算	53
4.4	ビデオシースルー型拡張現実感試作システムの構築	60
4.4.1	プロトタイプシステムの構築	61
4.4.2	入力画像の実時間歪み補正	62
4.4.3	実画像と仮想物体の合成	63
4.4.4	試作システムの実行結果	64
4.5	考察	67
4.6	結言	68
5.	結論	69
	謝辞	72
	参考文献	73

目 次

1	Cone Tree(参考文献 [RMC91] より引用)	11
2	ノード選択時の回転アニメーション	12
3	Spiral Tree	13
4	評価実験に使用した Spiral Tree	16
5	Spiral Tree のずれの距離が小さい場合	25
6	Spiral Tree のずれの距離が適切な場合	25
7	Spiral Tree のずれの距離が大きい場合	25
8	情報作業空間の一部のみが提示される場合	29
9	重畳表示環境システム構成図	32
10	重畳表示環境	32
11	実験 3 の設定：重畳表示環境における目標探索タスクの設定	34
12	実験 3 の結果：平均探索時間	36
13	実験 3 の結果：マウス操作による仮想平面の平均移動量	37
14	実験 4 の設定：ターゲット領域中心位置の候補 (初期位置 CRT 画面内の場合)	39
15	実験 4 の実験風景	41
16	実験 4 で用いた重畳表示環境の境界領域上の立方体	44
17	カメラパラメータ推定手法における各座標系間の関係	53
18	ビデオシースルー型拡張現実感試作システムのシステム構成図	61
19	ビデオシースルー型拡張現実感試作システムの処理の概要	62
20	ビデオシースルー型拡張現実感試作システムの実行例 (その 1)	65
21	ビデオシースルー型拡張現実感試作システムの実行例 (その 2)	66

表 目 次

1	実験 1 の結果：各視覚化手法での平均タスク達成時間と平均クリック数	18
2	実験 2 の結果：順序の有無による Spiral Tree でのタスク達成時間の違い	21
3	実験 2 の結果：順序の有無による Spiral Tree でのドラッグ数の違い	22
4	実験 2 の結果：順序の有無による Spiral Tree でのクリック数の違い	22
5	実験 4 の結果：各環境を用いた場合の平均タスク達成時間	42
6	実験 4 の結果：初期位置とターゲット位置毎の没入型を用いたときの平均タスク達成時間 (秒)	42
7	実験 4 の結果：初期位置とターゲット位置毎の重畳型を用いたときの平均タスク達成時間 (秒)	43
8	実験 4 の結果：没入型でのタスク達成時間を 1.0 としたときの重畳型での平均タスク達成時間の割合	43

1. 序論

1.1 情報作業空間と情報のブラウジング

現在、計算機はその名称が示すように「計算」のみを行うにとどまらず、電子化された文書・画像・音声など様々な情報に対する記録・編集・管理・検索・伝達といった様々な処理を行うものとして広く認識されている。計算機が様々な情報を利用するための機械として用いられるようになった背景には、電子化された情報の処理を行うための「作業空間」としてのユーザインタフェースの発展の歴史がある。

Xerox PARC(Palo Alto Research Center)で1970年代に開発された Smalltalk, その流れを汲んで1980年代に開発された Xerox Star, 現在の Macintosh の原型となった Lisa は, 計算機の画面上に机やフォルダといった現実環境のメタファ(比喩)を視覚的に導入した [CRM91, R. A86]. これらの研究で導入された概念はデスクトップメタファと呼ばれる。デスクトップメタファと WIMP(Window, Icon, Menu, Pointer) によるインタフェースは階層ファイルシステムという抽象的な概念をフォルダやウィンドウという具象的な図形に置き換えて表示することで計算機の専門家ではない一般ユーザが階層ファイルシステムを理解することを容易にした。また, 個々のファイルはアイコンとして表示されるようになり, 視覚的にファイルを識別できるようにするとともに, アプリケーションの起動は従来のコマンド入力 of の代わりに対応するアイコンをマウスを用いて選択することによって行なわれるようになった。ファイルの移動などもドラッグ&ドロップとよばれる直接操作によって行なわれるようになった。このように, 計算機内のデータやファイルといった情報の検索・管理にデスクトップのメタファを利用し, 図形として表現したファイルなどを操作することで情報を管理するインタフェースは現在の計算機のインタフェースの主流となっている。

その後, MIT における Dataland の研究ではデスクトップメタファの大型スクリーンによる拡大が試みられた [R. A86]. Card らがデスクトップメタファの概念を発展させて提案したルームメタファでは, 複数のデスクトップをタスク毎に切り替えること, 必要ならば複数のデスクトップ間で情報を共有することを可能

にした [CH87]. さらに, 文献 [CRM91] で, 計算機を利用するときの情報探索をより大きな情報処理のタスクの一部と捉えるべきであると主張した. その上で現実世界に実在する情報処理システムを観察し, 計算機を用いて情報を処理するためのユーザインタフェースに, 現実世界に実在する情報処理システムの性質を反映させた. これにより, その現実の情報処理システムが持つ利点を計算機における作業空間に実現しようとした. こうしてルームメタファを情報アクセスのための作業空間へと発展させたユーザインタフェースを「情報作業空間 (Information Workspace)」と定義した. Card らは提唱した情報作業空間の一実現例として 3次元ユーザインタフェースとアニメーションを利用した Information Visualizer を実装している [CRM91, MRC91, RMC91, RCM93, RPh⁺95].

情報作業空間を実現する上で重要な構成要素の一つに「情報視覚化 (Information Visualization)」が挙げられている [CRM91]. 情報視覚化とはファイルなどの情報間の関係をコンピュータグラフィックスを用いて視覚化することであり, これを用いることでユーザが情報間の関係を容易に理解できるようになる. 情報は視覚化されることにより, ブラウジングが可能となる. ブラウジングとは明確な目標を持たずに全体を通して眺める作業である. 情報を利用して用いる作業において, 「出版社 A の書籍 B」というようにいつも検索対象として具体的な目標がある訳ではなく, 「統計処理の入門書」というように漠然としか目標が定まっていない場合や, あるいはただ情報群を眺めていて自分の目的作業に関連がありそうな情報があれば取り出そうと考えている場合など, ユーザ自身が必要な情報が何であるのかを明確に表現できない場合も多い. このような場合には, ブラウジングにより引き出す情報を明確にするきっかけを得ることができる. 水野らは, 情報の検索時においては, 従来からのデータベースなどによる情報のフィルタリングと, 情報を様々な視点で鳥瞰するブラウジングが等しく重要であるとしている [水野 93].

以上の背景から, 情報を処理する道具としての計算機のインタフェースにおいて効果的な情報のブラウジングが可能な情報作業空間を提供する事は重要な問題であると考えられる. 情報を効果的にブラウジングする環境を提供するためには以下の 2 点を考慮する必要がある.

情報の効果的な視覚化 ブラウジングの対象となる情報を直感的に理解する事ができるように，わかりやすく視覚化する．

Fisheye View 効果 [Fur86] の提供 ユーザが注目している領域の周辺は詳細まで表示し，同時に，全体構造が把握できるようにする．

1.2 本研究の目的

本論文では，情報を効果的にブラウジングするための情報作業空間の構築を目標とする．そこで，目標となる情報作業空間構築のために

1. 順序付き階層構造の 3 次元視覚化，
2. 拡張現実感技術による情報作業空間の拡張，
3. 複数の特徴点を撮影した画像からカメラパラメータを推定する手法

の 3 手法を提案する．以下では各手法を提案する目的とその意義について順に述べる．

1.2.1 順序付き階層構造の 3 次元情報視覚化

本研究の第 1 の目的は，順序付き階層構造の 3 次元情報視覚化手法を提案することである．計算機を利用する上で視覚化されることが重要な構造のひとつに，ファイルシステムなどに採用されている階層構造が挙げられる．ブラウジングの対象となる情報群が階層構造として管理されていれば，ユーザは必要な情報が何であるのかを明確に意識していない場合でも，自分の興味に近そうなグループからブラウジングしていくことで，徐々に具体的な情報へと興味を絞り込んで行き，引き出す情報を具体化していくことができる．このように情報を属性や内容によって分類し階層的に管理することで，ユーザが情報を利用する際に必要な情報がどこにあるのかが分かりやすくなり，ブラウジングの負担を軽減することができる．このため，図書館の書籍の分類や UNIX のファイルシステムなど，情報は階層的に管理されることが多く，多くのアプリケーションで階層構造情報が利用されている．このことから，階層構造情報のブラウジング手法は重要で応用範

囲も広いと考えられており，階層構造情報を視覚化する研究が従来から幾つか知られている [JS91, Ren94, JR94, RMC91].

階層構造のような抽象構造を視覚化する手法の中でも，透視投影表示による 3 次元コンピュータグラフィックスを用いた 3 次元形状による視覚化手法は 3 次元視覚化と呼ばれ，

- 画面領域を効果的に利用して全体構造を視覚化する，
- 注目領域を周辺より大きく表示するという魚眼視野効果を自然に提供できる，
- 奥行き感覚によって知覚の手がかりが増加する

といった利点を持つ．これらは情報のブラウジングにとって重要な利点となる．

また，階層構造をブラウジングする際にユーザが利用する情報に階層内の属性によって付けられた順序情報が挙げられる．例えばファイル構造では「ファイル」という情報に対する「ファイル名」「ファイルサイズ」などの属性を元に階層内のノードに順序を与えることができる．各階層内のノードの属性に順序を付けた階層構造を順序付き階層構造と呼ぶ．この階層内順序を明示的に視覚化することはブラウジング時の手がかりを増加させることができるため，情報のブラウジングにとって重要であると考えられる．

以上のことから，本論文では，順序付き階層構造をブラウジングする視覚化手法として，3 次元視覚化手法を適用し，階層内の順序をブラウジング時に利用しやすいように明示的に視覚化する手法の構築を目指す．

1.2.2 拡張現実感技術による情報作業空間の拡張

本研究の第 2 の目的は拡張現実感技術によって情報作業空間を拡張する手法を実装し，評価することである．情報の 3 次元視覚化手法では情報を構成する単位となる情報を仮想物体で表現し，個々の情報間に存在する関係を仮想物体同士の 3 次元的な配置関係や色などの属性を用いて提示するものが多い．このときに，HTML のタイトルや，図書の書誌名を表す文字列をラベルとして仮想物体に付すことで仮想物体が表現する情報の内容をユーザに示すということが良く行われる [RMC91, MRC91, Ren94, 大隈 98]. このため，大規模情報を 3 次元視覚化す

るときには文字を解読するのに十分な解像度が要求され、情報作業空間の提示には CRT など高解像度のデバイスを用いることが多い。しかし、CRT などの実空間に固定された画面上にのみ 3 次元空間を投影する場合には CRT の画面範囲に投影可能な領域内しか一覽できず、視覚化される情報が莫大になったときに領域内が煩雑になり、視覚化された情報の構造の理解が困難になる。そこで、解像度を保ちつつ視覚化された情報を一覽できる領域を拡大する手法が必要となる。このため、

1. 大型スクリーンを用いる [R. A86].
2. 領域を提示している画面の周囲に多数の画面を配置し、表示領域を拡大する [CNSD93].
3. 透過型 HMD を用いて仮想空間を現実空間に投影するにより、現実環境を情報作業空間として利用する [FS91].

等の方法が提案されている。

大型スクリーンを使用した場合、投影面の面積は拡大されるが、投影面のない方向が存在するため、蓄積される情報量はそれほど大きくはならない。多画面式の手法では高い解像度を保ちつつ、仮想環境を拡大することができるが、装置が大掛かりになりすぎる。3 番目の手法のように計算機によって生成される仮想環境と現実環境を合成して提示する技術は一般に拡張現実感技術 (Augmented Reality) と呼ばれる [Azu97]。拡張現実感技術は、現実環境における作業を補助するための情報を仮想物体としてユーザに提示することが可能であることから、新たな情報提示手法の 1 つとして注目を集めている。たとえば、機械の保守手順の教示 [FMS93]、医療現場における手術中の患部情報の付加 [SLG⁺96] など、様々な応用が提案されている [FMHS93, RN95, SMR⁺97, GAB⁺98, 大島 98]。この技術を情報作業空間の提示に用いることで、視覚化された情報を現実環境に存在するように提示することが可能となり、作業空間の提示領域が拡張される。本研究では装置の規模や実用性から考えて、拡張現実感技術を情報作業空間へ適用する方法を採用する。

拡張現実感技術を情報作業空間に適用する研究については従来，ノートパソコンの画面の外側に HMD を用いて仮想物体で X window 環境のウィンドウ枠を表示することで情報作業空間を拡張する Hybrid User Interface が知られている [FS91]．しかしながら Hybrid User Interface ではユーザの姿勢を制限するなど多くの問題があった．また，現実のデスクトップに仮想的な情報を合成表示して情報作業環境を提供するものとして，Digital Desk [Wel93] が知られている．拡張現実感技術により情報作業空間の提示領域が拡大されるだけでなく，現実環境に実在する紙の書籍などの情報と電子化された情報を一つの環境内で同時に扱うことができるという利点も生じる．

本論文では Hybrid User Interface の概念を 3 次元の情報作業空間の提示に拡張し 3 次元視覚化された情報を CRT と HMD を併用して提示する環境を実装する．

1.2.3 実環境画像からの実時間カメラ位置推定手法

本研究の第 3 の目的は実時間で安定して視点位置を推定する手法を実現し，拡張現実環境を実現する上での位置ずれ問題を解決することである．拡張現実環境を情報作業空間の提示手法として用いた重畳表示環境では，現実環境に存在する CRT による提示領域と HMD による提示領域の位置ずれが問題となった．この問題は重畳表示環境特有の問題ではなく，拡張現実環境を実現する上で一般的に問題となる現実環境と仮想環境の位置ずれ問題である．

拡張現実環境における位置ずれ問題の解消は拡張現実感を実現するための重要な課題の一つに挙げられ，現在までも幾つかの手法が提案されている [Azu97, TGW⁺95, SLG⁺96, KV98]．特に，ビデオシースルー型の拡張現実環境は現実環境を撮影した画像上に位置合わせした仮想環境を合成して提示するという原理上，見ための位置ずれが起こらないことから医療などの正確な位置合わせを必要とする応用分野において注目されている．

本論文では従来からコンピュータビジョンの分野で研究されてきたカメラパラメータの推定手法を用い，ユーザの視点位置に取り付けられたカメラで撮影される実環境画像からカメラパラメータの実時間推定を行う．これにより拡張現実環境の位置ずれ問題の解決を試みる．

1.3 本論文の構成

本論文の構成は以下の通りである。2章では情報を3次元的に視覚化する手法について述べる。まず、従来から階層構造の3次元視覚化手法として知られるCone Treeについて概要とその利点について述べる。次にこの手法がブラウジングに利用される階層内の順序情報の表現が不十分であることを指摘し、Cone Treeの各階層を螺旋上に配置することにより順序を視覚化する、順序つき階層構造の3次元視覚化手法Spiral Treeを提案する。その上でSpiral Treeの有効性を評価するために行なった2つの比較実験について述べる。

3章では、情報作業空間の提示領域を拡大する手法について述べる。幾つかの手法について検討し、拡張現実感技術を用いて情報作業空間の作業領域を拡大する、重畳表示環境を提案する。重畳表示環境ではHybrid User Interfaceと同様に、CRTの外側の領域をHMDによる仮想物体でユーザに提示する。その後に試作した重畳表示環境を評価するための2つの実験について順に述べる。

4章ではCRT領域とHMD領域の間に生じる位置ずれ問題を解決するための視点位置推定手法について論じる。視点位置から撮影された実環境画像を入力として、コンピュータビジョンの分野で従来から研究されてきたカメラパラメータの推定手法を用いることで正確な位置合わせを行う。特に本研究では、複数のカメラパラメータ推定手法を組み合わせることで、実時間で安定してカメラパラメータを推定する手法を実現する。

5章以降では本論文をまとめ、今後の課題や展開について述べる。

2. 順序付き階層構造情報の 3次元視覚化

2.1 諸言

World Wide Web の急速な普及や電子図書館の実現などにより、図書館のような広い場所を用意しなくても、デスクトップ上で大規模な情報源からさまざまな情報を利用することが可能となってきた。しかし、利用可能な情報の大規模化と多様化は、ユーザが必要な情報を見つけ出すことを困難にしている。このため、キーワード検索を用いてユーザが検討する情報の候補の数を減少させることがよく行われる。しかし、キーワード検索で得られるすべての情報がユーザの興味に深く関連するとは限らない。また、ユーザが自分の興味をキーワードとして具体的に表現できない場合には、キーワード検索を用いての情報検索は困難となる。

そこで、ユーザが自分の興味に深く関連すると予想される情報を直感的に見分けて検討するためのブラウジング手法が望まれる。このため、大量の文書の間関係や組織の階層構造のような大規模情報の複雑な構造をユーザに分かりやすい形で視覚化する手法が注目を集めている。例えば、FilmFinder[AS94]ではスライドバーなどのグラフィカルユーザインタフェース(GUI)を用いてパラメータを動的に設定し、このパラメータを用いた検索結果を、実時間で視覚化することで、検索候補の素早い削減を可能にしている。また、Galaxy of News[Ren94]では階層的に分類された情報を仮想空間に提示する。初期状態ではおおまかな分類名のみが表示され、ユーザが興味を持つ分類名に視点を近付けることで、一つ下の階層に属する分類名が表示される。これを繰り返すことで、視点の移動による直感的なブラウジングを行うことができる。Information Visualizer[CRM91]は、3次元コンピュータグラフィックスとアニメーションを用いて複雑な情報の構造を視覚化し、ユーザに分かりやすく表示するシステムである。Information Visualizerでは、ブラウジングの対象となる情報を3次元的に視覚化することで、全体構造の直感的な理解を助けることを目指している。代表的なアプリケーションとして、線形情報を3次元視覚化するPerspective Wall[MRC91]、階層構造を3次元視覚化するCone Tree[RMC91]、参考文献間のリファレンスの関係を3次元視覚化する

る Butterfly visualizer[MRC95] などが知られている。

以上のように、さまざまな構造をもつ情報、もしくは情報の検索結果を視覚化することで、ユーザが必要な情報を捜し出すための手がかりを提供する手法が提案されている。本章では、視覚化される情報の構造として階層構造に注目する。階層構造は計算機のファイルシステムや電子図書館における書籍の管理など、大規模情報の管理によく用いられる構造であるため、階層構造情報をユーザに分かりやすく視覚化する手法は重要であると考えられる。

階層構造情報を 3 次元的に視覚化する手法としては Cone Tree が知られている [RMC91]。また、検索に用いるキーワードとの関連度を Cone Tree に反映する視覚化手法 [寺岡 96] や、1000 ノード以上の大規模な階層構造情報を視覚化しても画面が煩雑にならない様にするために、Cone Tree のノード配置方針の改良や色情報を利用する手法 [CK95]、枝の長さやノードの大きさにフラクタルレイアウトを適用する手法 [小池 94] など、Cone Tree を基に検索の手がかりを増大させる手法が提案されている。本章では、各階層内の情報の順序に着目し、Cone Tree のノード配置方針に階層内の順序を反映した手法 Spiral Tree を提案する [OTK⁺96, 大隈 95]。各階層内の順序情報は検索の手がかりとして階層構造のブラウジング時に頻繁に用いられるため、順序情報の明示的な視覚化はユーザのブラウジングに有効であると考えられる。

本章ではまず 2.2 節で従来の階層構造の 3 次元視覚化手法として知られる Cone Tree について述べたあと、階層内順序が利用しづらいという問題点を指摘する。2.3 節ではこの問題を解決するために、Cone Tree を改良して階層内順序を明確に視覚化する、順序付き階層構造の 3 次元視覚化手法 Spiral Tree を提案する。2.4 節では Spiral Tree の有効性を評価するために行った、Cone Tree と Spiral Tree の比較実験とその結果について述べる。続いて 2.5 節では Spiral Tree における順序の利用に関して行った被験者実験について述べる。2.6 節で Spiral Tree のノード配置方針や本手法の位置づけについて考察を述べた後、2.7 節でこの章をまとめる。

2.2 Cone Tree : 階層構造の 3 次元視覚化手法

Cone Tree による視覚化例を図 1 に示す. 図 1 中の画面に正対する長方形のプレートはノードと呼ばれ, 一つの情報であらわす. 半透明の 3 次元の円錐は情報の親子関係を視覚化したもので, 親ノードを円錐の頂点, 子ノードを円錐の底面の円周上に均等に配置する. この円錐を階層的に配置することで階層構造を視覚化する. すなわち, あるノード n の位置を \mathbf{p}_n , n の子が属する階層を視覚化する円錐の半径を r_n , 円錐の高さを l とし, n が m 個の子ノードを持つとき, n の i 番目の子ノードの位置 \mathbf{p}_i は式 1 のように表される.

$$\mathbf{p}_i = \mathbf{p}_n + \begin{pmatrix} l \\ -r_n \sin\left(\frac{i \times 2\pi}{m} + \alpha_n\right) \\ r_n \cos\left(\frac{i \times 2\pi}{m} + \alpha_n\right) \end{pmatrix} \quad (1)$$

但し, 円錐の軸上を頂点から底面に向かう方向を x 軸の正の方向とする. r_n, l の各々は, 階層構造全体が視覚化されたときにディスプレイの表示領域に収まるように計算して決定される. α_n はノード n の子の階層の回転角であり, ユーザのブラウジング時に動的に変化する.

このようにして視覚化された Cone Tree の任意のノードをマウスで選択すると, 選択されたノードは画面手前側にアニメーションを伴って移動する. 選択されたノードが画面手前にアニメーションする様子を図 2 に示す. 図 2 に示す様に各階層の円錐が独立に回転して, 選択されたノードからルートノードまでのパス上のすべてのノードが画面手前側に移動する. すなわち, ノード n の i 番目の子が選択された場合,

$$\frac{i \times 2\pi}{m} + \alpha_n = 0 \quad (2)$$

となるように α_n が変化する.

また, Cone Tree では必要に応じてマウスカーソルによるポインティングにより指示された階層以下についての表示/非表示を切り替えることができる. これは, ユーザにとって興味がない階層が他の階層をブラウジングするときの邪魔にならないようにするためである.

Cone Tree の特徴はアニメーションと 3 次元透視投影表示を積極的に利用して

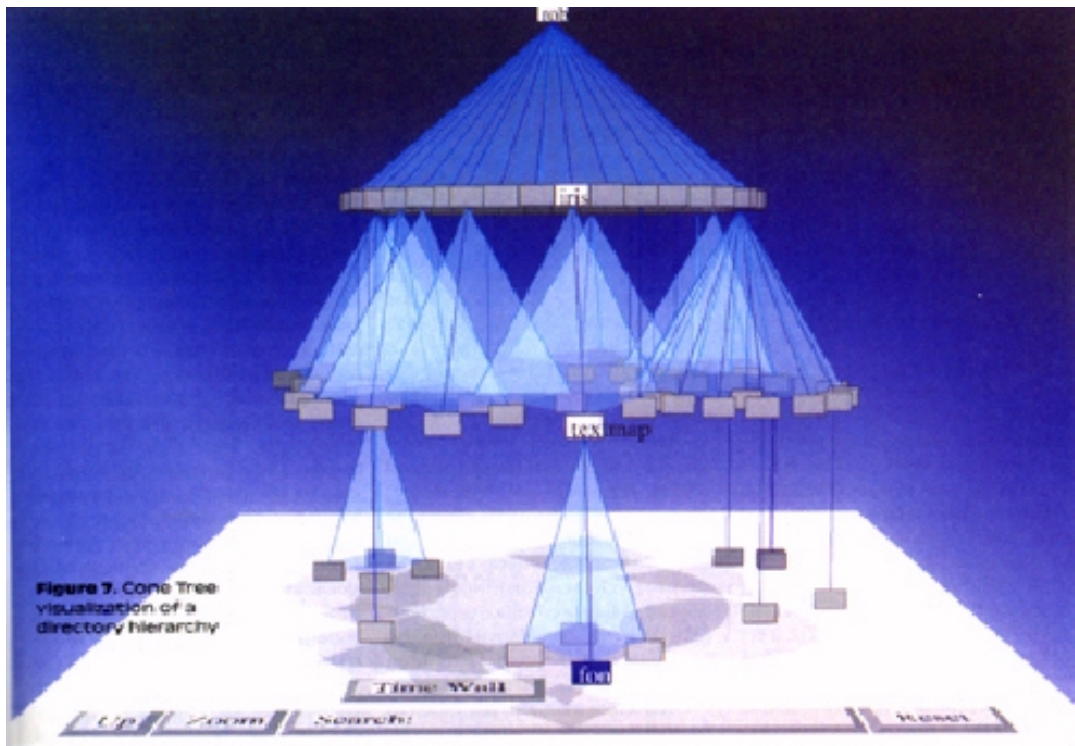


図 1 Cone Tree(参考文献 [RMC91] より引用)

いる点にある。アニメーションと 3 次元透視投影表示の利用については、それぞれ以下の利点が挙げられる。

- ノードが選択されたことによって起るノードの移動をアニメーションにより補間することで、ユーザが移動後の構造を再認識して心内で再構築する必要がなくなる。
- 透視投影による 3 次元的な表示は、視点に近い程画面に大きく表示されることになり、注目部分を詳細に表示し注目部分から遠ざかる程小さく表示するという魚眼視野効果を自然に実現する。

このように Cone Tree は階層構造のブラウジングにアニメーションと 3 次元透視投影表示をうまく適用しているが、ユーザが実際に階層構造のブラウジング

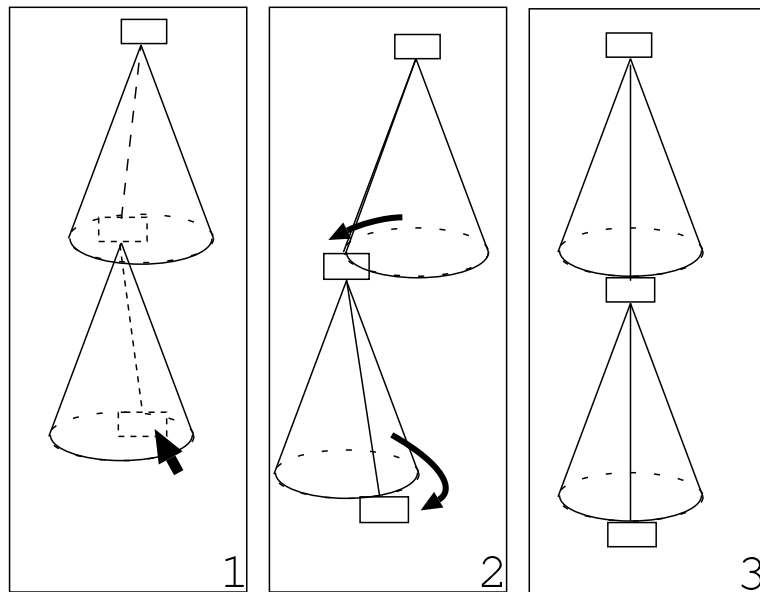


図 2 ノード選択時の回転アニメーション

を Cone Tree を用いて行う場合に、階層内の順序を利用しづらいという問題がある。階層内の順序とは、例えばファイル構造では「ファイル」という情報に対する「ファイル名」「ファイルサイズ」などの属性をもとにしてつけた順序のことである。各階層内の情報に順序を付けた階層構造を順序付き階層構造とよぶ。ユーザは、ある情報を検索するとき、この順序を手がかりに探索を行うことが多いと考えられる。しかし、Cone Tree では同一階層のノードを円周上に並べるので、各階層内の情報のある順序にしたがって整列しても、先頭と末尾の位置が分かりづらい。このため、ユーザは同一階層内のノードの順序を探索の手がかりとして十分に利用することができない。これは Cone Tree のノード配置方針が階層内の順序を十分に視覚化できていないことが原因である。よって、ユーザは順序付き階層構造情報の探索に不便を感じる可能性がある。

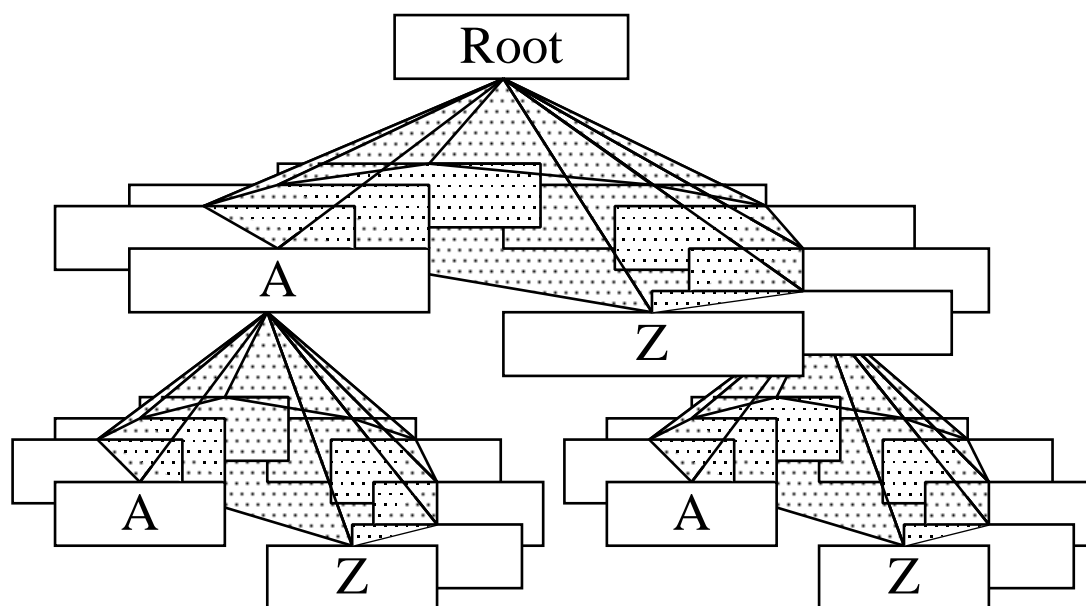


図 3 Spiral Tree

2.3 Spiral Tree : 順序付き階層構造の 3 次元視覚化手法

2.3.1 順序の利用

Cone Tree において情報探索過程で階層内の順序が利用しづらいという問題点は、各階層のノードを円周上に配置するという配置方針のために階層内の先頭ノードと末尾ノードの位置が明確でないことが原因である。そこで、階層内の先頭と末尾の位置を明示する手法について検討する。これには、

1. 末尾と先頭の間切れ目を入れる、
2. 先頭ノードの色を変えて表示する、
3. 各階層のノードを螺旋上に並べる、

などの方法が考えられる。これらの手法のうち、1の手法は Cone Tree の外見をほとんど変えることなく実現できるが、階層内のノード数が増えたときに切れ目が目立たなくなるため、ユーザに対して「切れ目を探す」という負荷を与える

ことになるという問題がある。2の手法については、順序関係を色空間で視覚化する最も有効な指針が明確でないため、実現が困難である。また、色情報は他の属性の視覚化に利用することも考えられる。そこで、本研究では3の方法を採用することとした。

各階層のノードを螺旋上に1周させる形で並べる方法で階層の先頭と末尾を明示するCone Tree型の階層構造視覚化手法をSpiral Treeとよぶ[OTK⁺96, 大隈 98]。Spiral Treeのノード配置の様子を図3に示す。

Spiral TreeはCone Treeに比べて、以下の利点を持つと考えられる。

- 螺旋配置により順序の利用が容易になる、
- 円錐の軸方向へのずれにより同一階層内のノード同士の重なりが減少するため、注目している階層についての一覧性が増す。

2.3.2 ノードの配置

Spiral Treeのノード座標はCone Treeのノード座標を、円錐の頂点から底面に向かう軸方向に平行移動して求められる。すなわち、ノード n の子ノードの先頭と末尾のノードの x 座標の差を d とし、ノード n の i 番目の子ノードの位置 \mathbf{p}_i は以下のようにして計算される。

$$\mathbf{p}_i = \mathbf{p}_n + \begin{pmatrix} l + \frac{i \times d}{m} \\ -r_n \sin\left(\frac{i \times 2\pi}{m} + \alpha_n\right) \\ r_n \cos\left(\frac{i \times 2\pi}{m} + \alpha_n\right) \end{pmatrix} \quad (3)$$

ここで、階層間距離 l に対してずれの距離 d が小さいと先頭と末尾の位置が明確になりにくい。 $(d = 0$ のときCone Treeとなる。) また、大きすぎるとSpiral Tree全体が画面内に収まらない上、見づらくなる。よってこれらの視覚化パラメータは適切な大きさに設定する必要がある。しかし、Spiral Tree全体がもっとも分かりやすくなる視覚化パラメータは視覚化の対象となる階層構造の1つの階層内のノード数や階層構造全体の深さなどにも依存し、全ての階層構造を分かりやすく視覚化する最適値を決定することは困難である。しかし、実際のSpiral Treeを用いるアプリケーションプログラムなどにおいては、これらの視覚化パラ

メータをユーザによってインタラクティブに変更可能にすることでこの問題を回避することができる。ずれの距離に関する議論は 2.6 節の考察でも述べる。

2.4 実験 1 : Cone Tree と Spiral Tree の比較実験

2.4.1 目的と設定

Spiral Tree のノード配置アルゴリズムが Cone Tree のノード配置アルゴリズムの改良になっていることを確認するために、同じ階層構造情報を視覚化した Spiral Tree と Cone Tree を用いて、指示された情報を見つけ出して選択する作業について、作業時間と操作数を比較した。

実験タスク 実験タスクにはノード探索タスクを用いた。ノード探索タスクとは以下の手順を 1 試行とする実験タスクである。

1. 被験者に目標ノードをフルパス名で指定する。(フルパス名とはルートノードから目標ノードまでのパス上に存在する各階層のノード名を“/”記号をはさんでつなげた文字列とする。)
2. 被験者は視覚化された階層構造を操作して目標ノードを選択（マウスでクリック）する。試行中も画面上部に目標ノード名を表示する。

ノード探索タスクはブラウジングによる情報探索作業を想定している。ノード探索タスクのタスク達成時間やタスク達成に必要な作業数は視覚化された階層構造情報の見やすさや操作性の影響を受けると考えられるため、これらを分析することで、視覚化された階層構造情報の見やすさや操作性について調べることができる。

実験手順 実験用 Cone Tree および Spiral Tree を使用して階層構造情報を被験者に提示した。実験で視覚化された階層構造情報は被験者の記憶力や背景知識の影響を受けないようにするために、階層構造としての意味を持たないものとした。ノードには順序をつけるための属性として名前を与え、同一階層内のノードは名

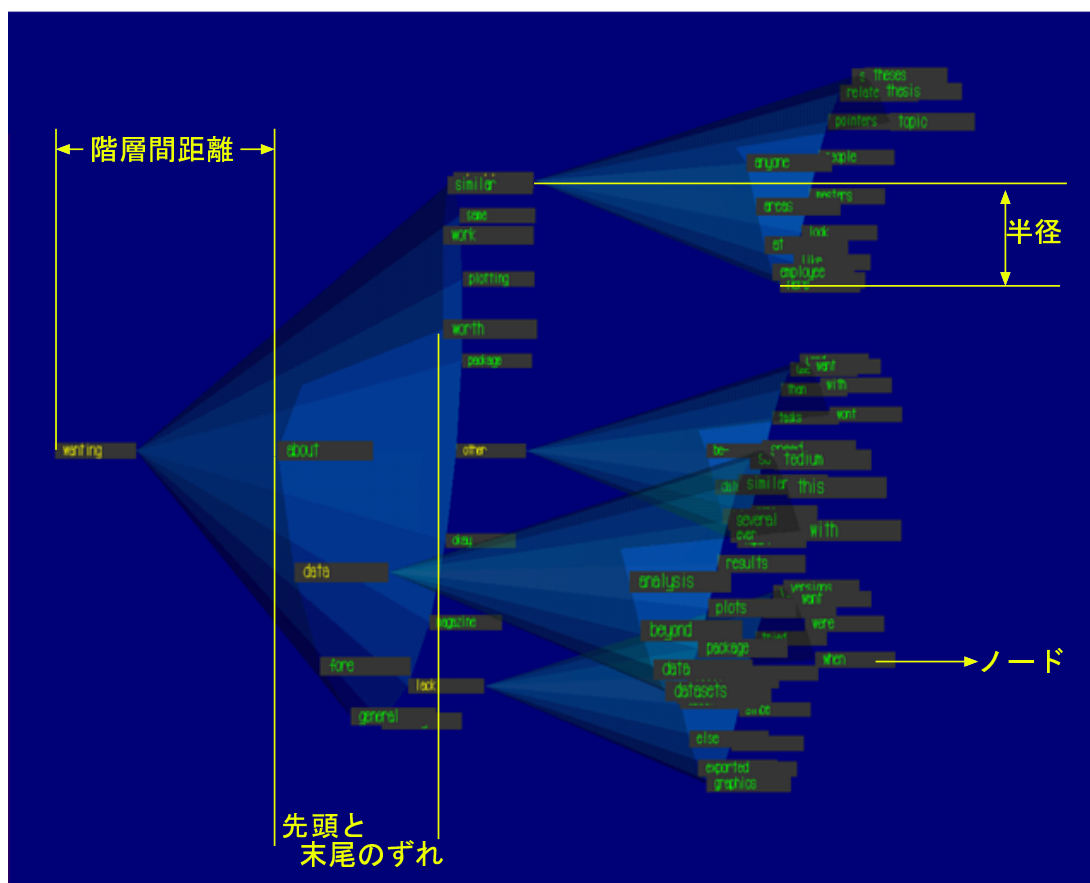


図 4 評価実験に使用した Spiral Tree

前によって辞書式順序で並べた。ノードに付した名前は英文から無作為に抽出した単語である。被験者はこの階層構造に対してノード探索タスクを行う。

被験者は両方の視覚化手法を用いて作業を行った。ノード探索タスクは、実験前に各々の視覚化手法を用いて 300 試行ずつ練習し、本実験では各々 100 試行ずつのデータを測定した。ノード探索タスクのターゲットノードはランダムに指定された。実験の際には、ノード探索タスク達成時間（目標が指定されてから選択するまでに要した作業時間）とノード探索タスクの達成に必要なマウスボタンのクリック数を測定した。また、実験後に被験者に対して Cone Tree と Spiral Tree のどちらを用いた方が検索が容易に感じたかを尋ねた。

実験環境 実験に用いた Cone Tree, Spiral Tree は横置き型とした。これは、通常の Cone Tree で全ノードのノード名を明記しようとするともノードの形が横長になるためノードの重なりが多くなり、ノード名が読みづらくなるからである。横置き型の Cone Tree は Cam Tree という別名で呼ばれる [RMC91]。しかし、本論文においてはこれも Cone Tree の名称で統一して呼ぶことにする。実験において、操作対象として視覚化した階層構造情報は以下のような構成であった。

- 1 層目 (ルートノードの子) : ノード数 16
- 2 層目 (ルートノードの孫) : ノード数 16 の階層が 4 つ

この階層構造は以下のパラメータで視覚化された。

- ノード : 縦 0.6cm × 横 3.6cm
- 階層間距離 l : 9.0cm
- 先頭ノードと末尾ノードのずれ d (Spiral Tree のみ) : 4.5cm
- 半径 一階層目 : 8.0cm, 二階層目 : 3.1cm
- 回転時のノードの移動速度 : 1cm / 秒

以上のパラメータを用いて実験に使用された Spiral Tree を図 4 に示す。

2.3.2 節で述べたように、全ての階層構造を分かりやすく視覚化する視覚化パラメータの最適値を決定するのは困難である。そこで上記のパラメータは次の手順で手動で決定した。まず、実験の準備段階で、ノード内の文字が読める程度にノードの大きさを決定した。次に、先頭と末尾が十分に明確になるように、階層内の先頭ノードの右端の x 座標よりも末尾ノードの左端の x 座標が大きくなるようにずれの距離 d を決定した。最後に、階層構造全体が画面内に納まるように半径と階層間距離 l を決定した。これらのパラメータでノード探索タスクが十分行なえることを予備実験により確認した上で本実験を行なった。

実験にはグラフィック WS (SGI 社 Onyx RE2), WS に標準装備のマウス, 21 インチディスプレイを使用した。被験者は本学の学生 6 名で、マウス操作に慣れており、階層構造の概念は理解していた。

2.4.2 結果

被験者 6 名のノード探索タスク達成時間を Cone Tree, Spiral Tree を使用した場合のそれぞれについて平均値を求めて比較した。結果を表 1 にまとめる。Spiral Tree を使用した場合の平均タスク達成時間は Cone Tree を使用した場合に比べて一秒近く短縮されている。これらのノード探索タスク達成時間の差に統計的な有意差が認められた。有意差の検定には分散分析 ($\alpha = 0.05$) を用いた。平均クリック数についても、ノード探索タスク達成時間と同様の分析を試みたが有意な差は見られなかった。

表 1 より、この実験において Spiral Tree を利用した場合、Cone Tree を利用した場合よりも、ノード探索タスク達成時間が減少することが統計的に認められることが分かる。この結果は、Spiral Tree による視覚化は Cone Tree による視覚化よりもノード探索時間の短縮（作業効率の向上）に有効であることを示している。

また、被験者の主観として「Spiral Tree と Cone Tree のどちらを用いた方が検索が容易に感じたか」の問に対しては全被験者が Spiral Tree の方が容易であると答えている。

以上の結果から、Spiral Tree は Cone Tree よりも、

1. ノード探索作業時間の短縮に有効、

表 1 実験 1 の結果：各視覚化手法での平均タスク達成時間と平均クリック数

	Spiral Tree	Cone Tree
平均タスク達成時間 (秒)	5.68	6.57
分散	1.49	2.92
分散分析 ($\alpha = 0.05$)	有意差あり ($P \ll 0.01$)	
平均クリック数	1.34	1.38
分散	0.043	0.064
分散分析 ($\alpha = 0.05$)	有意差なし ($P = 0.24$)	

2. ノード探索作業の操作感の向上に有効,
であると考えられる.

2.4.3 考察

被験者の主観から, Cone Tree よりも Spiral Tree の方がノードの探索における操作感の向上に有効であったと考えられる. しかしながら Spiral Tree は Cone Tree に比べて階層としてのまとまりが認識しにくいと報告した被験者もいた. この意見は Spiral Tree の欠点を指摘している可能性がある.

また, この実験において使用された実験用の Spiral Tree と Cone Tree の実装上の不満として, 以下の3点が挙げられた.

- 階層を自由な角度から眺めることができない,
- 回転させたくない階層(親の階層)も回転する,
- 文字が読みづらい,

これらの不満点は, 実験用 Spiral Tree, Cone Tree に共通であるため, 実験結果に及ぼす影響は少ないと考えられる.

実験結果はノードの配置方針の違いから生じるものであるが, 順序の利用が可能になったことのみによるものかは分からない. これは Spiral Tree の配置方針では順序の利用を容易にする効果と, 注目階層の一覧性を向上させる効果の両方が期待されているからである. このため順序の利用に関する, より詳細な検討を行う必要がある.

以上の考察に基づき, 実装上の不満点を解消した実験用プログラムを用いて, 順序の利用に関する評価実験を試みた. 実験用プログラムに施した改善は以下の3点である.

- 階層の回転角度を自由に操作できるようにするために, ノードをマウスにより回転軸に垂直な方向にドラッグすることで, ドラッグされた量に応じて階層を回転させる.

- 操作の対象としている階層のみを回転させることにし，親の階層以上は回転させない.
- 文字はフォントを読みやすいものに変更する.

2.5 実験 2 : 階層内順序の利用に関する評価実験

2.5.1 目的と設定

この実験の目的は Spiral Tree で階層内の順序が利用されていることを確認することである. これは, 実験 1 から考えられる Spiral Tree の利点は

- ノードの重なるの減少による一覧性の向上
- 順序の利用

の両方の特徴に影響を受けていることを考慮し, ブラウジングの手がかりとしての順序が Spiral Tree で利用可能かを確認するためである.

実験環境 以下に示す階層構造 A, 階層構造 B を, 順序無し Spiral Tree (階層内の順序をランダムにした Spiral Tree) と通常の Spiral Tree で視覚化した.

2 種類の階層構造データを用いた理由は, 視覚化の対象となる階層構造に実験結果が大きく依存しないことを確認するためである.

- 階層構造 A
 - 1 層目 (ルートノードの子): ノード数 30
 - 2 層目 (ルートノードの孫): ノード数 10 の階層が 3 つ
- 階層構造 B
 - 1 層目 (ルートノードの子): ノード数 30
 - 2 層目 (ルートノードの孫): ノード数 10 の階層が 9 つ

被験者は本学の学生 5 名であった. なお, 本実験の被験者は実験 1 には参加していない.

実験手順 階層構造 A, 階層構造 B を順序無し Spiral Tree と通常の Spiral Tree で視覚化した場合の 4 通りそれぞれについて, ノード探索タスク 10 試行を被験者に行わせた. この 10 試行を 1 セッションとし, 1 回の実験で 10 セッションを行った. 実験は練習 2 回 (20 セッション), 本試行 3 回 (30 セッション) 行った. 各回の間隔は半日から一日であった. 記録したデータはタスク達成時間 (ノード探索タスク達成時間), ドラッグ数 (タスクの達成までにドラッグを使用した回数), クリック数 (タスクの達成までにクリックによる自動アニメーション使用した回数) である.

2.5.2 結果

階層内に与えられている順序を有効に利用できているかを調べるために, 順序が使用できる場合とできない場合についてのタスク達成時間, ドラッグ数, クリック数について結果を比較する. 表 2 にタスク達成時間, 表 3 にドラッグ数, 表 4 にクリック数の結果を示す.

まず, それぞれの階層が順序通りに並べられている Spiral Tree を用いた場合と, 順序通りに並べられていない Spiral Tree を用いた場合での, ノード探索タスクのタスク達成時間, ドラッグ数, クリック数のそれぞれについて, 統計的な有意差を検定した. 統計的な有意差の検定には分散分析 ($\alpha = 0.05$) を用いた. こ

表 2 実験 2 の結果 : 順序の有無による Spiral Tree でのタスク達成時間の違い

階層構造 A に対する順序の有無	順序なし	順序あり
平均タスク達成時間 (秒)	7.30	6.44
(分散)	(10.98)	(10.79)
有意差	なし ($P = 0.08$)	
階層構造 B に対する順序の有無	順序なし	順序あり
平均タスク達成時間 (秒)	13.25	12.63
(分散)	(44.52)	(43.14)
有意差	なし ($P = 0.49$)	

の際、タスクの難易度が目標ノードの存在する階層の深さに影響を受けることを考慮して、分析に使用するデータは目標ノードが2階層目に存在する場合に限った。この結果、階層構造Bを操作対象とするときのクリック数を除いて、統計的な有意差は認められなかった(表2~4)。

次に、それぞれの場合の平均値に注目する。表2~4のそれぞれについて、順序が利用できる場合と利用できない場合の平均値を比較すると、階層構造Aを対

表3 実験2の結果：順序の有無によるSpiral Treeでのドラッグ数の違い

階層構造Aに対する順序の有無	順序なし	順序あり
平均ドラッグ数(回)	0.49	0.32
(分散)	(0.49)	(0.32)
有意差	なし ($P = 0.09$)	
階層構造Bに対する順序の有無	順序なし	順序あり
平均ドラッグ数(回)	1.14	1.04
(分散)	(1.53)	(1.36)
有意差	なし ($P = 0.52$)	

表4 実験2の結果：順序の有無によるSpiral Treeでのクリック数の違い

階層構造Aに対する順序の有無	順序なし	順序あり
平均クリック数(回)	0.15	0.18
(分散)	(0.34)	(0.31)
有意差	なし ($P = 0.69$)	
階層構造Bに対する順序の有無	順序なし	順序あり
平均クリック数(回)	0.25	0.04
(分散)	(0.63)	(0.04)
有意差	あり ($P = 0.01$)	

象とするときのクリック数を除いて、タスク達成時間、ドラッグ数クリック数のすべてについて階層内順序を利用できる場合の方が平均値が減少している。

この結果は、Spiral Tree を用いてのノード探索タスクでは階層内順序を利用できる場合の方が、タスク達成時間やタスク達成に必要な作業数を減少する傾向を示している。このことは、Spiral Tree では階層内の順序が利用されている可能性が高いことを示す。

2.5.3 考察

実験 2 から新たに提供したドラッグによる階層の回転が多く用いられており、クリックはほとんど用いられていない。このことから、ドラッグによる階層の回転は被験者に好んで使用されたことが分かる。しかしながら、ドラッグによる回転は被験者が行う操作の自由度をあげるため、結果として、タスク達成時間の分散を大きくしたと考えられる。このことが、順序が利用できる場合と利用できない場合でのタスク達成時間に関する統計的な有意差が認められなかった原因の一つであると考えられる。

実験 2 では 2 種類の階層構造を用いた。これは、結果が視覚化の対象となる階層構造に依存しないことを確認するためであった。この 2 種類の階層構造に関して、タスク達成時間やドラッグ数については構造に依存せず、順序がある方がタスク達成時間などが減少する傾向にあることを観察することができた。しかし、この 2 種類の検討だけでは不十分であり、より複雑な階層構造を用いた場合についても検討を重ねる必要がある。しかしながら、すべての種類の階層構造に関して検討することは不可能であり、視覚化対象となる階層構造の複雑さがノード探索タスクに与える影響を厳密に調べるのは困難である。

2.6 考察

本節では、Spiral Tree を利用するとき決定が難しいと考えられる視覚化パラメータの設定について考察した後に、順序付き階層構造視覚化手法の研究における Spiral Tree の位置づけについて考察する。

前節での 2 つの実験では、Spiral Tree におけるずれの距離や階層間距離、各階

層の半径といった視覚化パラメータをあらかじめ調整した上で実験を行った。しかし、これらの視覚化パラメータが不適切に設定されればかえって見づらくなる可能性もある。例えば、図 5 に示す場合のように、階層間距離に対してずれの距離 d が極端に小さいと先頭と末尾の位置が明確になりにくく、図 7 に示すようにずれの距離が大きすぎると Spiral Tree 全体が画面内に収まらない上に階層としてのまとまりが見づらくなることが考えられる。また、半径の大きさも階層としてのまとまりやノードの重なりなどに影響を与えるパラメータであると考えられる。実験 1 において被験者が「階層としてのまとまりが認識しづらい」と報告したことや、実験 2 において階層内順序の有無に関する統計的な有意差が認められなかったことなどは、これらのパラメータが最適な値に設定されていなかった可能性を示している。

Spiral Tree では Cone Tree とは違い、階層の深さの違うノード同士の重なりが起こる可能性がある。例えば、図 7 に示すように、階層間距離に対してずれの距離が大きく設定された場合など、視覚化パラメータが適切に設定されていない場合には、ある階層内の先頭付近のノードの子孫と末尾付近のノードが重なって見えることがある。また、視覚化パラメータが適切に設定されていても、深い階層を持つ階層構造を視覚化した場合、祖先として各階層の先頭ノードばかりを持つ階層と末尾ノードばかりを持つ階層では、ずれの距離の蓄積によって、違う深さの階層が重なってしまう。このことは、実験で用いたような小規模な階層構造ではあまり問題にならないが、階層が大規模な場合には画面を繁雑にする可能性がある。しかし、大規模な階層構造の視覚化に関しては、Cone Tree においても階層数においては 10 階層程度、ノード数においては 1000 ノード程度が限界であることが指摘されている。これより大規模な階層構造への対応は今後の課題として残されるが、Cone Tree で扱える 10 階層程度の規模の階層構造においてはずれの距離を階層間距離の $1/10$ 未満にすることで階層の深さが違うノード同志の重なりは回避できる。また、各階層のずれが蓄積されないように、各階層における先頭ノードの深さをそろえるように、階層間距離を変更する配置方針によっても、この問題を解決できる。

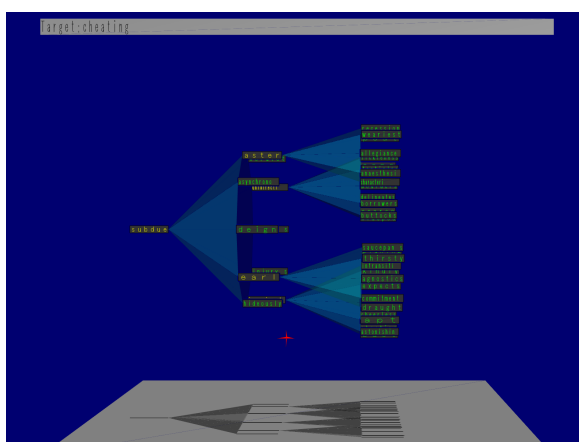


図 5 Spiral Tree のずれの距離が小さい場合

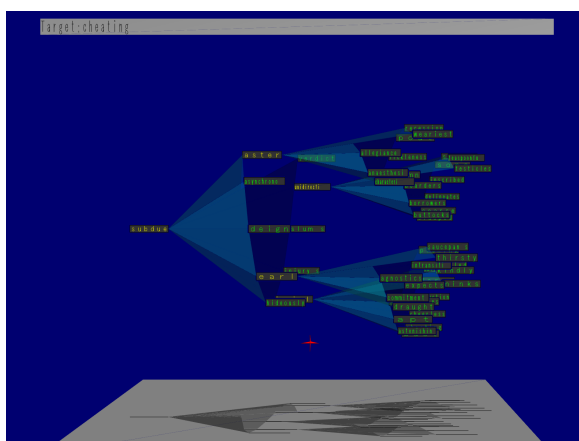


図 6 Spiral Tree のずれの距離が適切な場合

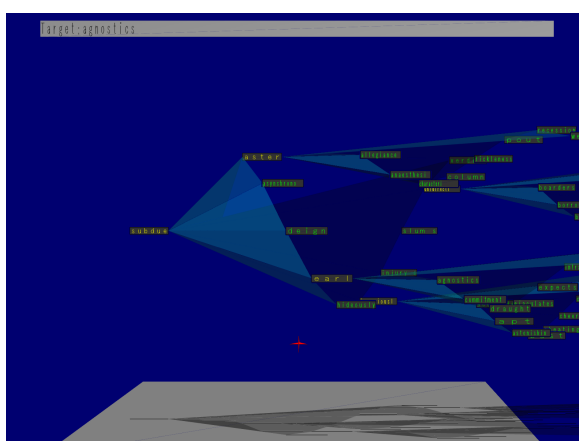


図 7 Spiral Tree のずれの距離が大きい場合

順序付き階層構造情報の視覚化問題全体に対する Spiral Tree の位置づけを検討するためには、2次元視覚化手法と3次元視覚化手法の比較や、あらゆるノード配置方針を検討する必要がある。本章で実施した実験からだけでは視覚化手法全体における Spiral Tree の有効性などについて論じることは難しい。また、他に考え得る全ての視覚化手法との比較は現実的に不可能である。しかしながら、本章で提案、実装した Spiral Tree は順序付き階層構造情報の3次元視覚化手法において被験者実験により既存の3次元視覚化手法に対する有効性を証明された一手法として位置付けることは可能であると考えられる。

2.7 結言

本章では大規模な情報を利用するための順序付き階層構造情報の表示手法として、従来の Cone Tree の改良型である Spiral Tree を提案した。

提案手法 Spiral Tree が順序を利用する検索に有効であることを確認するために、ノード探索タスクを用いた比較実験を行った。実験の結果、Spiral Tree を用いた場合の方が Cone Tree を用いる場合に比べてタスク達成時間や、タスク達成に必要な作業数を減少することが分かった。これは Spiral Tree が Cone Tree に比べて注目階層の一覧性が高く、階層内の順序を利用することができるという特徴によると考えられる。

また、Spiral Tree で実際に階層内順序が有効に利用されているかを確認するための実験も行った。階層内の順序を利用できる Spiral Tree と利用できない Spiral Tree を用いてのノード探索タスクを用いた比較実験の結果、順序が利用できる場合の方がタスク達成時間などが減少する傾向が観察され、Spiral Tree では順序が利用されている可能性が高いことを示した。

今回検討した Cone Tree 型の視覚化ではノード数で 1000、階層数で 10、階層の分岐の数が 30 程度より大規模な階層構造に適用した場合には画面が繁雑になるという問題が Robertson 自身により指摘されている。この問題について、提示ノード数を増加させるための従来研究 [CK95] との統合、仮想物体提示領域の拡大による密集の回避や、ユーザの使用状況に応じて変化するノードの配置方針など [寺岡 96] が知られているが、本研究では拡張現実環境を情報作業空間として

利用することにより，提示領域そのものを拡大し，ノードの密集を防ぐとともに，現実環境に存在する情報を同時に利用することを考える．次章では拡張現実感による 3次元情報作業空間について述べる．

3. 拡張現実感による 3 次元情報作業空間の拡大

3.1 諸言

前章で扱ったような情報を 3 次元視覚化する手法では、情報を構成する単位となる情報を仮想物体で表現し、個々の情報間に存在する関係を仮想物体同士の 3 次元的な配置関係や色などの属性を用いて提示するものが多い。このときに、HTML のタイトルや、図書の書誌名を表す文字列をラベルとして仮想物体に付すことで仮想物体が表現する情報の内容をユーザに示すということが良く行われる [RMC91, MRC91, Ren94, 大隈 98]。このため、大規模情報を 3 次元視覚化するときには文字を解読するのに十分な解像度が要求される。

しかし、CRT などの実空間に固定された画面上にのみ 3 次元空間を投影する場合には CRT の画面範囲に投影可能な領域内しか一覽できない。その上、限られた領域内のみを提示領域として用いると、視覚化される情報が莫大になったときに領域内が煩雑になり、視覚化された情報の構造の理解が困難になる。より大きな仮想空間を情報作業空間として用い、固定され限定された領域内にその一部を提示する手法には

- 全体の構造を一覽できず理解しにくい、
- 視点を移動するための自然な手法を提供する必要がある、
- 視点を移動するうちに自分の現在位置を把握するのが困難になり迷子になりやすい

等の問題が存在する。

以上のような背景から、3 次元視覚化された莫大な情報を提示するためには、解像度を保ちつつ視覚化された情報を一覽できる領域を拡大する手法が必要とされる。

本章ではまず 3.2 節で情報作業空間を提示する領域の拡大手法に関する適用可能な従来手法について述べ、評価のために実装した重畳表示環境について述べる。3.3 節では実装した重畳表示環境を用いて重畳表示環境の利点を確認するために

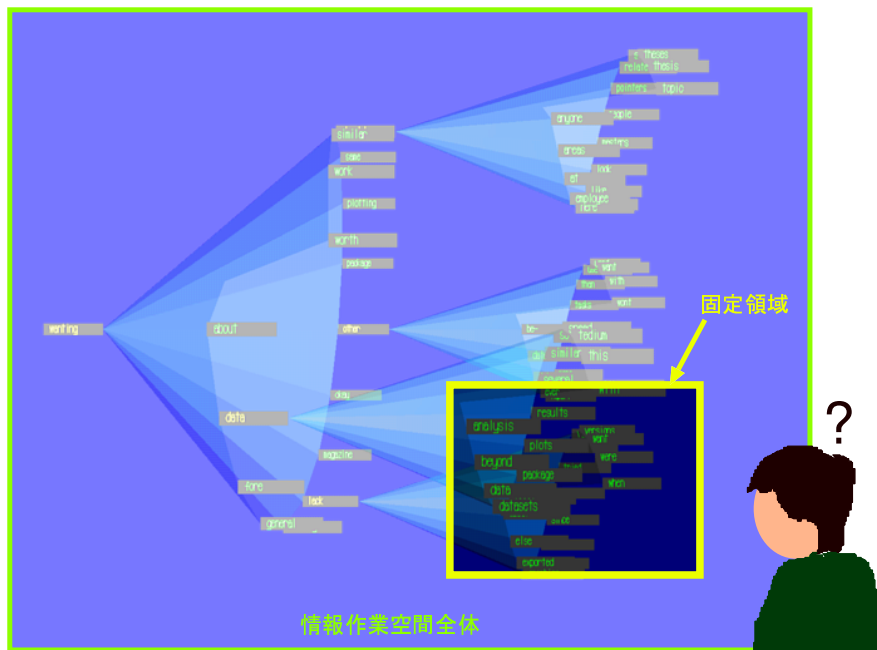


図 8 情報作業空間の一部のみが提示される場合

行った実験と、その結果について述べる。3.4 節では重畳表示環境の現状で問題となる境界領域のずれについて考察し、続く 3.5 節でこのずれがユーザの作業に与える影響を調べるために行った実験について述べる。最後に 3.6 節でこの章をまとめる。

3.2 情報作業空間提示領域の拡大手法

解像度を保ちつつ視覚化された情報を一覧できる領域を拡大するために、大型スクリーンを使用した場合 [R. A86] には、投影面の面積は拡大されるが、投影面のない方向が存在するため、提示可能な領域はそれほど拡大されない。これに対して、ユーザを取り囲むように多数の画面を配置することで視野角を確保することができる多画面式的手法 [CNSD93] では高い解像度を保ちつつ、仮想環境を拡大することができるが、装置が大掛かりになる。通常の画面の領域の外を透過型 HMD を用いて仮想空間として提示する手法 [FS91] では多画面式の方法に比

べて、透過型 HMD に提示される領域での解像度の面で劣るが、装置の規模は大きくなる。

文献 [FS91] で採用されている手法のように、計算機によって生成される仮想環境と現実環境を合成して提示する技術は一般に拡張現実感技術と呼ばれる。拡張現実感技術は、現実環境における作業を補助するための情報を仮想物体としてユーザに提示することが可能であることから、新たな情報提示手法の 1 つとして注目を集めている [Azu97]。たとえば、機械の保守手順の教示 [FMS93]、医療現場における手術中の患部情報の付加 [SLG⁺96] など、様々な応用が提案されている [FMHS93, RN95, SMR⁺97, GAB⁺98, 大島 98]。拡張現実感技術を情報作業空間の提示手法として用いることで、視覚化された情報を現実環境に存在するように提示することが可能となる。本研究では装置の規模や実用性から考えて、拡張現実感技術を情報作業空間へ適用する方法を採用する。

Feiner らは文献 [FS91] において、デスクトップメタファのインタフェースを採用しているノートパソコンを用いて、液晶画面の外側の領域もデスクトップとして使用するために透過型 HMD を用いて液晶画面の外側のデスクトップを提示する Hybrid User Interface を提案した。Hybrid User Interface は、ユーザの視点位置を固定し、ユーザの視点を中心としてノートパソコンの液晶画面を含む半球面上を、デスクトップとして使用するものであった。しかしながら、液晶画面の外側の領域に提示されるのはデスクトップメタファにおけるウィンドウの枠のみであり、ユーザの視点が半球の中心から外れるとウィンドウ枠が正確な位置に表示されないなど、大きな制限があった。

また、拡張現実感技術を情報作業空間へ適用した他の従来手法としては、現実のデスクトップにプロジェクタを用いて仮想環境を投影し、カメラからの画像を用いてユーザの手やペンで仮想物体とのインタラクションができる環境である Digital Desk [Wel93] が知られている。Digital Desk では現実の環境に対して仮想環境を投影するので、現実環境に実在する紙の書類と電子的な書類を同時に扱えることが最大の利点となっている。デスクトップメタファを拡張現実環境を用いて提示する利点は情報提示環境の提示方法として用いる場合の利点にもそのまま当てはまる。

そこで本章では、CRT など提示領域が限定された高解像度の提示デバイスの外側を透過型の HMD を用いて没入型の仮想環境を提示することによって情報作業空間を提示する環境について考える。即ち、Hybrid User Interface の概念を 2 次元のデスクトップメタファから 3 次元情報作業空間に拡張することで、CRT と透過型 HMD という性質の違う 2 種類の提示デバイスによって一つの仮想空間をユーザに提供する環境である。この環境は拡張現実環境でもあるため、現実環境に存在する情報も、電子的な情報と同時に利用できる。このような環境を重畳表示環境と定義する。

本論文において試作した重畳表示環境は以下の方法で実現する。

1. 磁気トラッカで測定した視点の位置と視線方向を HMD, CRT の画面に投影する際の透視投影変換の理論上の視点と一致させる。
2. CRT の画面が存在する位置を正確に設定しておき、視点と CRT 画面からなる四角錐 (CRT に投影する際の視体積) を遮蔽する仮想平面を HMD に表示する。

これにより、解像度の高い CRT による提示領域と一覽性の高い透過型 HMD による提示領域を同時に利用することが可能となると考えられる。この設計にしたがって重畳表示環境を試作した。

試作したシステムの構成を図 9 に示す。視点、ポインティングデバイスの三次元位置を磁気トラッカを用いて計測し、グラフィックワークステーションに入力する。ポインティングデバイスに取り付けられているスイッチの ON/OFF の状態はグラフィックワークステーションのシリアルポートを通して取得できる。ユーザは透過型 HMD を装着し、ハーフミラーを通してグラフィックワークステーションに付属の 21 インチ CRT を観察する。透過型 HMD と CRT の画面に提示する仮想環境の描画用に 2 台のグラフィックワークステーションを用いた。2 台のグラフィックワークステーションは LAN で接続されており、視点とポインティングデバイスの三次元位置を送受信する。HMD 領域を描画するグラフィックワークステーションは DA スキャンコンバータを通して透過型 HMD に画像を出力する。

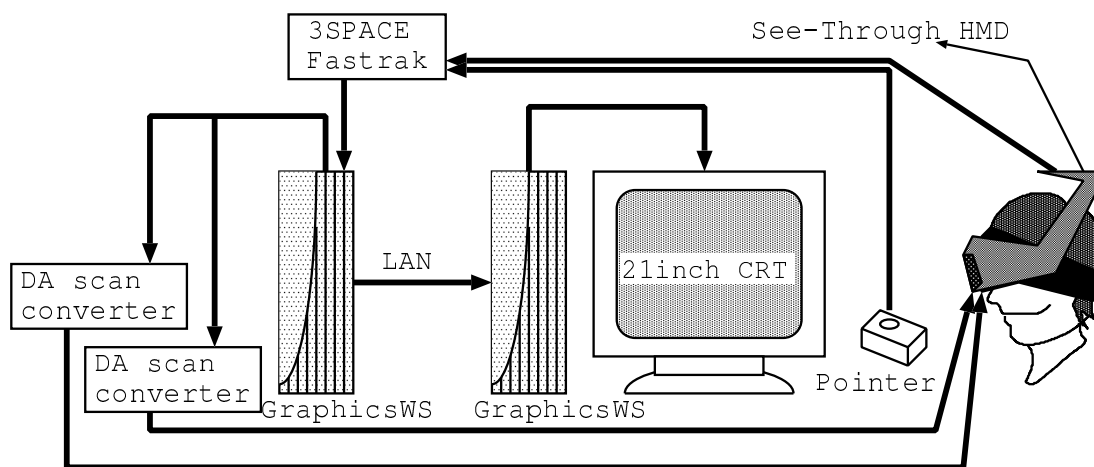


図 9 重畳表示環境システム構成図

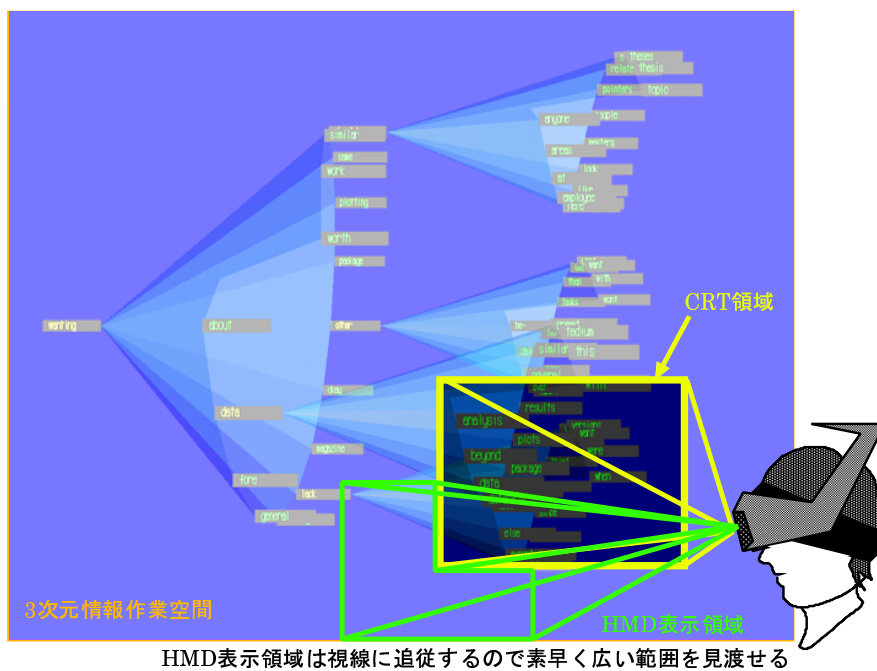


図 10 重畳表示環境

3.3 実験 3：目標探索タスクによる評価実験

3.3.1 目的と設定

重畳表示環境は 2 種類の提示デバイスを同時に利用することで解像度が必要な作業と広い提示領域を見渡す必要がある作業の両方を効率良く行えるように設計されている。ユーザが 2 つのデバイスを必要に応じて利用することで、重畳表示環境の利点をうまく利用できるかを確認するために CRT のみ、HMD のみ、重畳表示環境の 3 種類の提示手法の比較実験を実施した。

実験手順 実験は以下の手順を繰り返すことで進められた。図 11 に実験中に提示される環境を示す。

1. 図 11 に示すように、100 個の指標図形 (ターゲットの円 : 1 個, ダミーの正方形 : 99 個) が 1 辺 100cm の正方形内の 10×10 の格子上に配置され、被験者に提示される。円の位置は格子上の 100 点からタスク毎にランダムに決定される。
2. この格子上に指標が付けられた仮想的な平面は CRT 平面に重なるように配置される。被験者はこの平面をマウス操作によって移動させながらターゲットを探索する。
3. 図 11 に示すように、CRT 中央部には、1 辺 10cm の正方形の枠が提示されており、被験者はこの領域に発見したターゲットを移動させてマウスボタンをクリックする。

以上の設定で

- ターゲットの直径およびダミーの正方形の一辺の長さが 1cm, 2cm, 3cm の 3 通りの場合、
- 提示する環境が CRT のみ、HMD のみ、重畳表示環境の 3 通りの場合

の組合せによる合計 9 通りの場合で実験タスクを行なわせ、探索タスク達成に要した時間と、探索タスク達成までに仮想平面を移動させた量を計測した。被験者

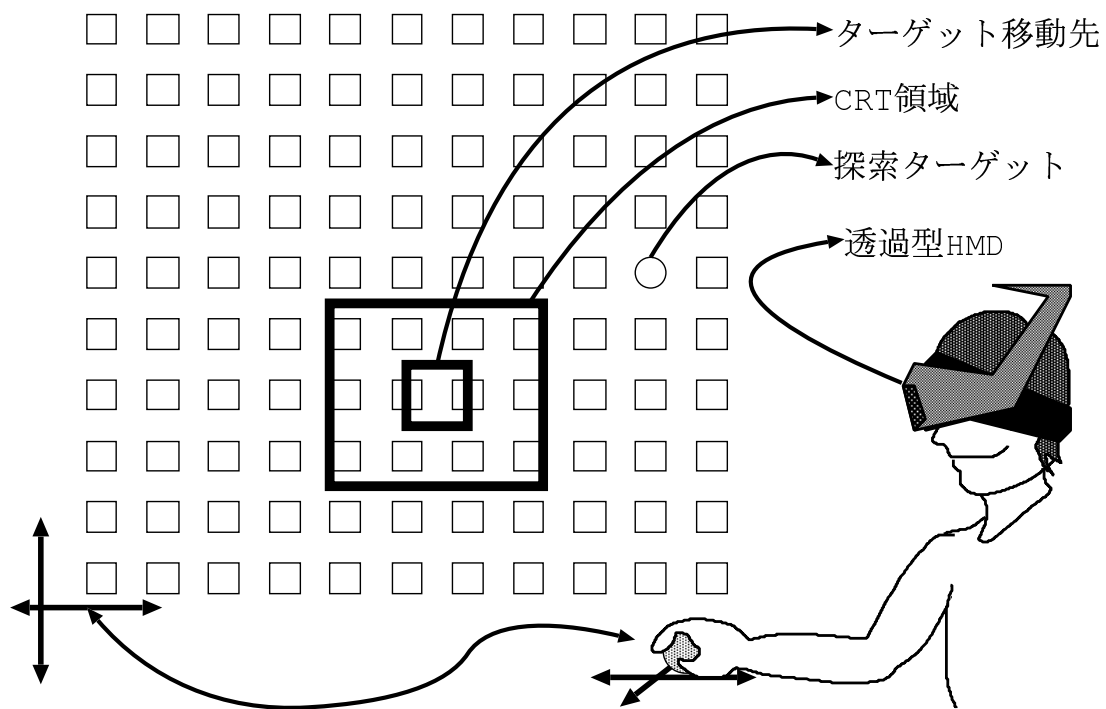


図 11 実験 3 の設定：重畳表示環境における目標探索タスクの設定

は本学の学生 2 名であり，マウス操作に習熟していた．タスクの試行数は設定毎に一人あたり 100 試行とした．

3.3.2 結果

以下では，まず CRT のみ，HMD のみを利用した場合の結果について検討し，これを踏まえて重畳表示環境の結果を検討する．

図 12 に示す平均探索時間のグラフより，探索時間については以下のことが分かる．

- 指標が小さい場合には，CRT のみによる提示手法を用いる場合にターゲットの探索時間が短くなっている．
- 指標が大きく HMD の解像度でも十分判別可能なときは，HMD を用いる方

が早くターゲットを見つけることができる。

また、図 13 に示すマウス操作によって仮想平面を移動させた量については、以下のことが分かる。

- HMD を用いる場合は仮想平面移動量が小さい。
- CRT を用いる場合は仮想平面移動量が大きい。
- 指標の面積によらず全体として HMD のみを用いる方が平面の移動量が小さい。即ち、指標の面積の影響より表示環境の影響が大きい。

ここで重畳表示環境を用いる場合に注目する。指標が大きい場合には HMD と同様の方法を用いて全体を見渡すだけでターゲットを見つけることができるので、探索時間、平面の移動量は共に HMD を用いた場合と同様に小さい。さらに、指標が小さくなってターゲットの判別に解像度が必要になる場合にも CRT 領域を用いて探ることができるため、CRT のみの場合と同等の早さで探索することができる。仮想平面の移動量も、サイズが小さいときは CRT 領域を利用し、サイズが大きくなると HMD 領域を用いて探索したことを示している。

以上の議論から重畳表示環境では CRT 領域、HMD 領域を必要に応じて使い分けることができ、CRT、HMD 両者の特徴を活かして作業環境をうまく拡張していると考えられる。

3.3.3 考察

被験者からの意見として、HMD 使用時の見回し遅延による位置ずれの影響が重畳表示環境においてより顕著に感じられるということが挙げられた。見回し遅延による位置ずれとは視点位置が測定されてから仮想環境が描画されるまでの処理時間により、実際のユーザの視点位置よりも描画された仮想環境が想定している視点位置が遅れることにより発生する位置ずれを示す。HMD のみを利用している場合には利用する作業環境全体が見回し遅延の影響を受けてずれのため遅延を感じても作業に支障はないが、重畳表示環境では CRT 領域に見回し遅延が発生しないため、境界でずれを感じてしまう。このずれによる違和感は実験結果には反映されていないが、重畳表示の現状での問題点を示している。

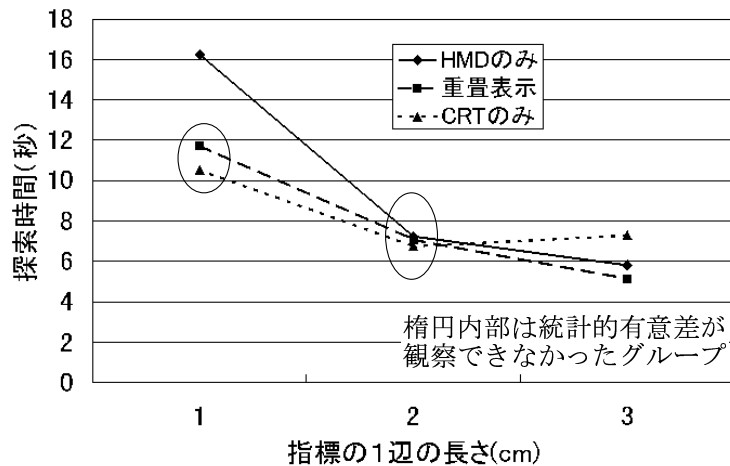


図 12 実験 3 の結果：平均探索時間

3.4 実装上の問題点

前節での実験の考察で述べた，見直し遅延による位置ずれの問題は拡張現実環境を利用する際の本質的な問題の一つである．拡張現実感技環境を構築する上で位置ずれが起こる原因は以下のように分類できる．

静的パラメータの計測誤差 ユーザの両眼間隔やユーザの視点と HMD の投影面の正確な位置関係，投影面の正確な大きさなどは一度計測すれば良いパラメータであるが完全に正確な計測は不可能である．この計測誤差が位置ずれの一つの一つの原因となる．

動的パラメータの計測誤差 仮想現実感技術や拡張現実感技術で用いられる 3 次元位置計測センサの測定誤差である．電磁気式の 3 次元センサの磁場の歪みによる誤差などがこれにあたる．

通信遅延などによるずれ 3 次元位置計測センサの値が計算機に入力されてから描画結果がユーザに提示されるまでの間に，ユーザの視点位置は実際には

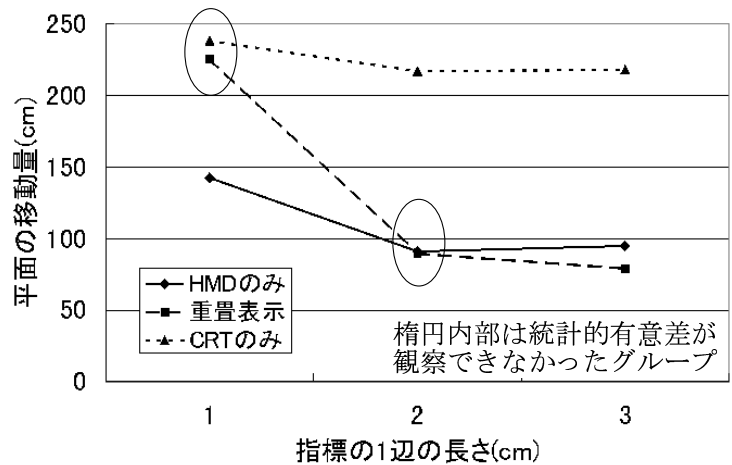


図 13 実験 3 の結果：マウス操作による仮想平面の平均移動量

すでに変化している．このためある時刻におけるセンサの計測値と実際のユーザの視点位置にはずれが生じる．この時間的なずれは見回し遅延としてユーザに知覚される．

重畳表示環境の場合，これらの原因による現実物体と仮想物体の位置ずれは，「現実物体」である CRT 画面に描画される「仮想物体」と HMD に描画される「仮想物体」のずれとして知覚される．その結果，仮想物体同士がずれて表示されることになり，通常の仮想現実環境や拡張現実環境よりも，違和感をうける可能性がある．このような位置ずれを減少させる手法については，仮想現実感，拡張現実感の分野で幾つかの研究が成されているが [OT96, WO95]，全ての問題を完全に解決するには至っていない．

また現在，試作した重畳表示環境の提示に使用しているデバイスに起因する問題点として，HMD と液晶シャッター眼鏡の併用が困難であることが挙げられる．本来重畳表示環境は 3 次元情報作業環境を提示するためのものであり，両眼立体視を用いるほうが望ましい．液晶シャッター眼鏡と HMD の併用が可能となれば，

重畳表示環境全体を両眼立体視可能になる。しかし、現在は HMD 領域にしか両眼立体視を可能な仮想物体を提示できない。

以上の位置ずれと立体表示に関する問題点が重畳表示環境を利用する際に双方の領域間の仮想物体の移動時に違和感を与えて、移動作業に影響を与える可能性がある。重畳表示環境においては仮想物体を双方の領域間で移動することは頻繁に起こると考えられる。例えば以下は重畳表示環境内で大規模情報を利用する際にユーザが行うと想定される作業の手順である。

1. 視覚化された大規模情報の全体構造の一覧を透過型 HMD の表示領域で行い、全体をブラウジングする。
2. 特に興味のある情報を取り出して CRT による表示領域に移動させて検討する。場合によっては編集する。
3. 検討し終って詳細を見る必要がなくなった情報は HMD 領域に移動させる。

このように、仮想物体の双方の領域間での移動は重畳表示環境においてユーザが良く行うと考えられる作業である。このような作業を行う場合に、2つの表示領域を用いても統一した環境として扱えるのであれば、通常の HMD のみによる没入型の仮想環境と同様の作業時間で仮想物体の移動を完了できると考えられる。そこで、重畳表示環境のずれがどの程度ユーザの作業に影響を与えるのかを、比較実験により確認した。次節ではこの実験の詳細を述べる。

3.5 実験 4：仮想物体移動タスクによる評価実験

3.5.1 目的と設定

重畳表示環境における 2つの表示領域のずれがユーザの作業にどの程度の影響を与えるかを検証するために比較実験を実施した。本実験では、重畳表示環境内における CRT 領域と HMD 領域の間を仮想物体を移動させるという作業を被験者に行わせる。この作業にかかる時間を、透過型 HMD のみを用いて同じ作業を行った時間と比較することで、2種類の表示領域を併用したときに、仮想物体の移動時間などがどの程度影響を受けるかを調べた。表示領域のずれによって違和

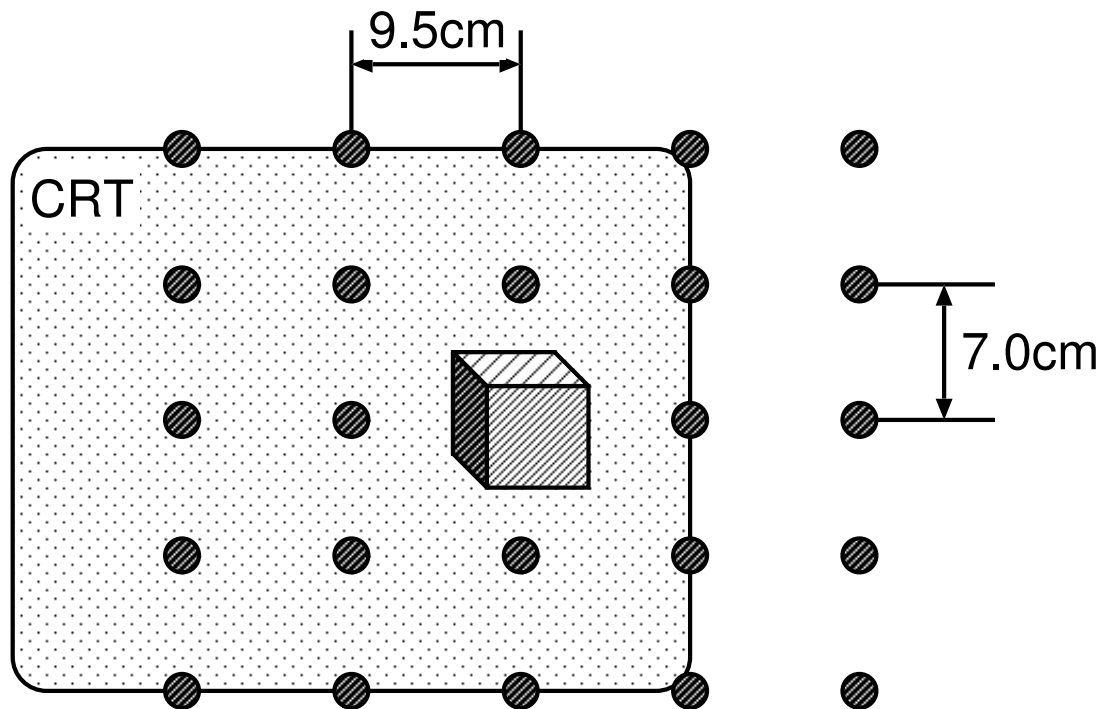


図 14 実験 4 の設定：ターゲット領域中心位置の候補 (初期位置 CRT 画面内の場合)

感が生じる場合，特に 2 つの提示領域の境界領域に対する作業に影響が現れると考えられるので，境界領域に注目して分析を行う。

実験タスク 本実験で用いた仮想物体移動タスクとは，仮想環境内に被験者が移動させる一辺 5cm の不透明な立方体 (移動物体) と移動先となる 1 辺 7cm の半透明な立方体 (ターゲット領域) を配置しておき，3 次元カーソルを用いて被験者にその仮想物体をターゲット領域へ移動させるという作業である。

初期位置の移動物体を 3 次元カーソルで把持してからターゲット領域内に包含された状態で置かれるまでを 1 試行とし，24 試行で 1 セッションとする。それぞれのセッション毎に移動物体の初期位置は固定されている。ターゲット領域の中心位置は移動物体の初期位置を中心とする 5×5 の格子点上に試行毎に配置され

た。ただし、全格子点の中心(移動物体の初期位置)以外の点とした。1セッションで中心を除く全ての格子点が1度ずつターゲット領域の中心に設定されるものとし、出現順序はランダムなものとした。図14に移動物体の出現位置の格子点の配置を示す。格子点間の距離は横が9.5cm、縦が7.0cmであり、これはCRT画面の幅と高さをそれぞれ4等分した距離とした。

設定 本実験では、仮想物体移動タスクを、重畳表示環境と透過型HMDのみを用いた環境との2種類の仮想環境のもとで行なわせ、タスク達成時間を比較した(図15)。

重畳表示環境を用いた場合に、移動物体の初期位置が

- CRT中央の点を右へ9.5cm平行移動した点(CRT表示領域),
- CRT中央の点を右へ19.0cm平行移動した点(領域境界),
- CRT中央の点を右へ28.5cm平行移動した点(HMD表示領域),

に設定された3セッションについてこの作業にかかる時間を調べた(実験4-1)。同様に、透過型HMDのみを用いて移動物体とターゲット領域が実験4-1と同じ場所に配置された場合の作業時間を調べた(実験4-2)。被験者は実験4-1の3セッションと実験4-2の3セッションの合計6セッションを行なった。行なうセッションの順番は被験者ごとに変更した。被験者は8名であった。キャリブレーションは被験者毎に行い、実験中に視点が移動する範囲内での領域間のずれが1cm程度以内に収まるように設定した。

実験中、初期位置に配置された移動物体を持ってからタスク終了までの時間(タスク達成時間)を記録した。また、実験後に被験者にどのようなときに思い通りに操作できなかったかなどの意見や感想を求めた。

仮想物体の操作に用いる3次元カーソルの提示位置は手に持った入力デバイスに取り付けられた磁気トラックの位置からCRT画面へ下ろした垂線に沿ってCRT方向へ15cm平行移動した位置とする。これはCRT領域に提示するとき手が3次元カーソルを隠して見にくくならないようにするためである。



図 15 実験 4 の実験風景

3.5.2 結果と考察

分析には、被験者 8 名のタスク達成時間から最大値と最小値を取り除いた中間 6 値の平均値 (平均タスク達成時間) を用いた。

まず全体の傾向を見るために、6 通りのセッションそれぞれにおける平均タスク達成時間と、没入型でのタスク達成時間に対する重畳表示環境でのタスク達成時間の比を表 5 に示す。没入型ではすべての仮想物体が HMD に表示されるが、表 2 では移動物体の初期位置が重畳表示環境ではどの領域に表示されるかで分類している。表 5 より初期位置が CRT 領域、領域境界、HMD 領域の全ての場合において、タスク達成時間は重畳型を用いると没入型を用いたときの 1.3 倍以上、全

平均でも約 1.5 倍になっている。

次に，初期位置が CRT 領域，領域境界，HMD 領域の 3 通りと，ターゲット領域が CRT 領域，領域境界，HMD 領域の 3 通りの組合せによる 9 通りの場合についてのそれぞれの平均タスク達成時間を表 6，表 7 に示す。

表を見ると，HMD のみを用いた没入環境でも移動物体の初期位置とターゲット領域の中心位置の組合せでタスク達成時間が変化している。これは移動物体の初期位置からターゲット領域までの平均の距離がそれぞれの場合で異なることが原因であると考えられる。そこで，表 7 のそれぞれの項目の時間を表 6 の対応する項目の時間で割って正規化を行う。この結果を表 8 に示す。

表 8 から，タスク達成時間はもっとも悪い場合で 2.2 倍程度であることが分かる。また，没入型仮想環境に比べて重畳表示環境でタスク達成時間が 1.5 倍以上に増加しているのは以下の 4 通りの場合であることが分かる。

表 5 実験 4 の結果：各環境を用いた場合の平均タスク達成時間

初期位置	CRT	境界	HMD	全平均
没入型	2.57	2.02	2.05	2.21
重畳型	3.37	3.41	2.92	3.23
重畳型/没入型	1.31	1.69	1.42	1.46

表 6 実験 4 の結果：初期位置とターゲット位置毎の没入型を用いたときの平均タスク達成時間 (秒)

	ターゲットの中心位置		
初期位置	CRT	境界	HMD
CRT	2.40	2.65	2.63
境界	1.91	1.91	2.19
HMD	2.15	1.98	2.07

- CRT 領域から領域境界への移動
- 領域境界から CRT 領域への移動
- 領域境界から領域境界への移動
- HMD 領域から CRT 領域への移動

まず，ターゲット領域の中心位置が領域境界にある場合に注目する．移動物体の初期位置が CRT 領域，領域境界である 2 つの場合に特に大きくタスク達成時間が増加するという結果は，被験者から「境界に配置された仮想物体が非常に分かりにくい」という報告を多く受けた事と関連があるように思われる．つまり，移

表 7 実験 4 の結果：初期位置とターゲット位置毎の重畳型を用いたときの平均タスク達成時間 (秒)

初期位置	ターゲットの中心位置		
	CRT	境界	HMD
CRT	2.63	4.10	2.93
境界	3.34	4.22	2.80
HMD	4.04	2.81	2.74

表 8 実験 4 の結果：没入型でのタスク達成時間を 1.0 としたときの重畳型での平均タスク達成時間の割合

初期位置	ターゲットの中心位置		
	CRT	境界	HMD
CRT	1.10	1.55	1.11
境界	1.75	2.21	1.28
HMD	1.88	1.42	1.32

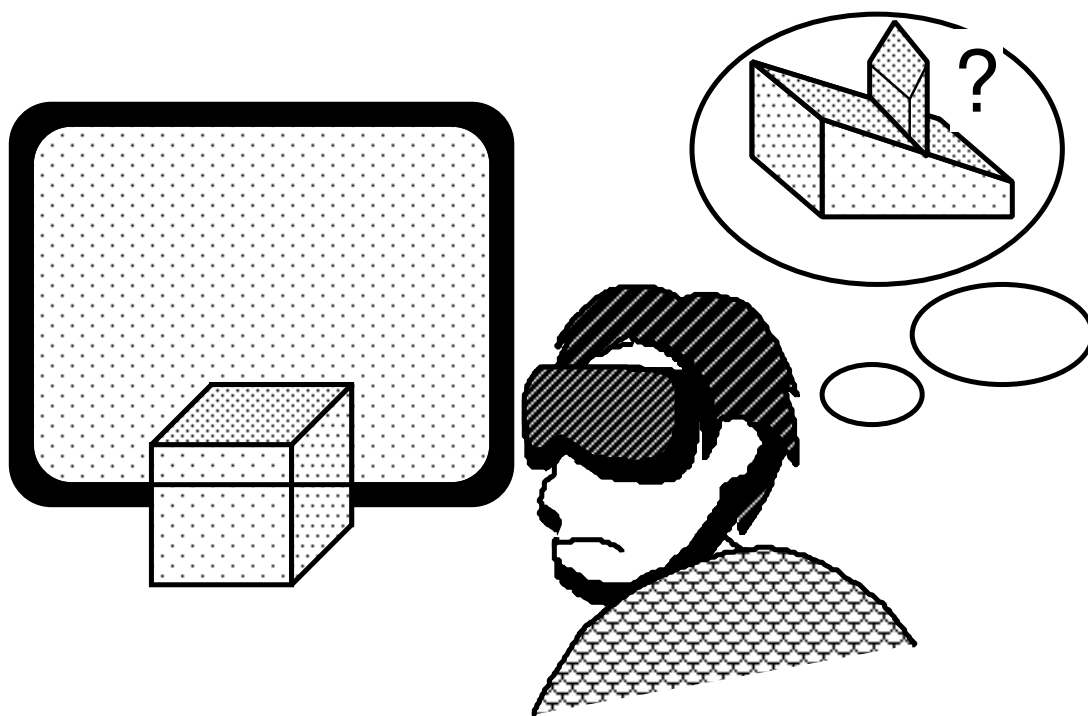


図 16 実験 4 で用いた重畳表示環境の境界領域上の立方体

動物体やターゲット領域を示す立方体は、図 16に示すように、一部が立体的に、残りが平面的に見えるために立方体として認識するのが困難となり、タスク達成時間に影響を与えたと考えられる。これに対して移動物体の初期位置が HMD 領域内にある場合には、前述の 2つの場合よりタスク達成時間に対する影響が少ない。これは初期位置において既に移動物体を立体として認識しているので、ターゲット領域が領域境界に存在しても、ターゲット領域の HMD 領域に表示される部分だけを手がかりに配置することができたことによるものと考えられる。

ターゲット領域の中心位置が CRT 領域にある場合にも、移動物体の初期位置が、領域境界上にある場合と HMD 領域上にある場合の 2つの場合でタスク達成時間が大きく増加している。このことは「重畳環境使用時には CRT 領域内が平面的に表示されるので、立方体の奥行きが分からず、ほとんどハイライトによる色情報に頼って作業した」という被験者の報告などから、単眼視による奥行き情

報の欠落が原因であると考えられる。

また、「重畳環境使用時には CRT 領域に対して配置する際に手が画面を隠して邪魔に感じた」という意見も CRT 画面内のターゲット領域への移動に時間がかかる原因に関連すると考えられる。実際には、カーソルと手に持つ入力デバイスとの間には 15cm のオフセットが付けられていることから、視点を移動することで手が移動物体やカーソルを隠すことを回避できる。しかし被験者は、「普段計算機を使う時には CRT を真正面からしか眺めないで別の角度からのぞき込むということを不自然に感じる」と報告した。

ただし、移動物体の初期位置とターゲット領域の中心位置がともに CRT 領域にある場合にはほとんどタスク達成時間は増加していない。これは、移動物体の初期位置における大きさを基準にして、ターゲット領域での奥行きを判断できるためだと考えられる。

ターゲット領域の中心位置が HMD 領域にある場合には、タスク達成時間はそれほど大きな影響を受けていない。これは HMD 領域への移動時には大きな違和感を感じずに重畳表示環境を使用できることを示していると考えられる。

以上より、以下の 2 点が明らかになった。

- HMD 領域と CRT 領域の境界で立体視の手法が異なると違和感の原因となる。
- それぞれの領域の境界のずれにより、最大で 2 倍程度の作業時間を必要とする。

これらの問題点は CRT 領域が両眼立体視できないことに起因するものであると考えられる。両眼立体視できる領域である HMD 領域にターゲット位置が設定されている場合には、タスク達成時間は特に大きな影響を受けていないことから、CRT 領域を両眼立体視できるようにすることができれば、特に大きな違和感を感じることなく 2 つの領域を使用できるようになると考えられる。

3.6 考察

文献[CRM91]で、Cardらは情報作業空間における作業のコスト構造(cost structure)を定義し、情報を用いる作業全体のコストを低くすることを目標に挙げている。重畳表示環境などの提示手法における視覚化された個々の情報の「コスト」は、視覚化された情報に付されたラベル文字列の読みにくさ(解像度の低さ)と捉えることができる。そこで情報作業空間としての重畳表示環境にのコスト構造を当てはめて考えると、CRTによる提示領域ではコストが低く(解像度が高く)、あまり多くの情報が蓄積できない直接蓄積領域(immediate storage area)に相当すると考えられる。外側のHMDによって新たに拡大された領域は、多少コストは高くなっても、多くの情報を蓄積することができる2次蓄積領域(secondary storage area)に相当するように設計されており、コスト構造の観点からも重畳表示環境は有効な設計であると考えられる。

また、現在は液晶シャッター眼鏡とHMDの併用ができないため、CRT領域の両眼立体視ができない。このため奥行き的位置合わせを必要とする直接操作による仮想物体操作は困難であると考えられる。CRT領域とHMD領域で提供する立体感が異なる場合に、形状がうまく把握できないのも問題となる。そこで、現在の段階では、

- Spiral Treeなど奥行き的位置合わせ操作を必要としないアプリケーション、
- HMD領域もCRT領域の延長平面のみを情報作業空間として使用し、3次元視覚化された物体はこの平面への透視投影表示で示される、

という適用法が実用的であると考えられる。

将来、デバイスの小型化などに伴ってHMDと液晶シャッター眼鏡の併用が可能になるなどして重畳表示環境全体で両眼立体視可能な仮想物体を提示できるようになった場合には3次元的操作を必要とするアプリケーション等にも問題なく適用できるようになると考えられる。

3.7 結言

本章では、大規模情報を利用するための仮想環境の提示手法として、CRT による提示領域を透過型 HMD を用いた没入型仮想環境を用いて拡大する重畳表示環境を提案した。これにより文書などの情報の検討や編集を行うための高い解像度と、多くの情報を配置できる広い仮想環境を同時に利用することができると思われる。この重畳表示環境を評価するために 2 つの実験を実施した。

1 つ目の実験ではターゲットを探索する作業を CRT のみ用いる場合、HMD のみ用いる場合、重畳表示環境を用いる場合の 3 通りでおこなって比較した。その結果、粗い解像度で十分な大きなターゲットを探索する場合には、HMD を用いることで素早くターゲットを探索することができ、ターゲットが小さい場合には、CRT を利用する方が探索時間を短縮できることが実験的に示された。提案してきた重畳表示環境ではその両方の利点を利用することにより、どちらの場合でも短い時間で探索することができていることが確認された。

また、2 つ目の実験の結果 2 つの領域の境界に起こるいくつかのずれが違和感を与え物体配置の作業の効率に与える影響を確認する実験を行なった。この実験には重畳表示環境を実際に使用する場合を想定した、CRT による表示領域と HMD による表示領域の間で仮想物体を移動させるタスクを用いた。この結果、2 つの領域間の境界では立体を正しく認識するのが困難になって仮想物体の配置に時間がかかることや、立体視ができない CRT による表示領域では仮想物体の奥行きが分かりにくくなって仮想物体の配置に時間がかかるという問題点を確認した。

今後の課題として、CRT による表示領域の両眼立体視の実現と、このときの仮想物体移動タスクへの影響の調査、文字列がラベル付けされた仮想物体の分類タスクによる重畳表示環境と HMD のみを用いた仮想環境表示との比較実験などが挙げられる。また、実装上の改善案として、平面的な表示領域では HMD 側にワイヤフレームによるバウンディングボックス等を表示することで奥行きに関する情報を提供することなどが挙げられる。また、これらの機能を実装した後に、今回と同様の評価実験を行うことも今後の課題として挙げられる。

4. 画像からの視点位置推定による実環境と仮想環境の幾何的整合

4.1 諸言

前章では、拡張現実感技術を3次元情報作業空間を提示するために利用することにより CRT に提示される高解像度領域と、HMD に提示される見回し可能な仮想空間を同時に使用する手法を提案した。しかし、実環境と仮想環境のずれが実用上の問題となっていたことから、本章では情報作業空間を提示するための拡張現実環境における位置ずれ問題の解決手法について検討する。

拡張現実環境における実環境と仮想環境の位置ずれは、主にユーザの視点位置姿勢の測定誤差が原因である。拡張現実環境は

1. 実環境に設定される基準座標系におけるユーザの視点の位置姿勢情報を獲得しておき、
2. 獲得した視点位置姿勢情報に基づいて描画された CG 画像を現実環境に合成して提示する

という手順で実現する。このため、ユーザの視点の位置姿勢情報の誤差が現実環境と仮想環境の位置ずれに反映されることになる。従来、拡張現実感システムにおいてユーザの視点情報を獲得する手法として採用されてきたものは、

1. 電磁気式、超音波式などの3次元位置センサを用いる手法 [FMS93, SLG+96]、
2. 現実環境をカメラによって撮影した画像から視点情報を推定する手法 [NC96, KV98, 大隈 97]

に大別できる。

電磁気式、超音波式の3次元センサは仮想現実感の分野で良く利用されており、安定して視点の位置・姿勢を計測できるという利点を持つ。しかし、電磁場や超音波の発生装置と計測装置(コイルやマイク)の対を用いて測定する機構で実現されているため測定範囲が電磁場や超音波の発生装置周辺に限られる。また、周

囲に金属が多く電磁場が歪みやすい環境、超音波を反射する物体がある場所などでは測定誤差も大きくなる。

画像から視点情報を推定する手法は、カメラが視点位置に取り付けられていると仮定し、入力画像内に撮影される3次元位置が既知の複数の点(特徴点)の画面座標からカメラパラメータ¹を推定することで視点情報を得る。この種の手法は特徴点の配置や認識方法を工夫することで測定可能範囲を自由に設定できる。しかし、特徴点を認識する処理はノイズや光源環境の変化などの影響を受けやすい。また、視点の移動にともなって、視点情報の推定に用いる特徴点が隠蔽されたり撮影範囲外へ出たりするため、カメラパラメータの推定に必要な特徴点を安定して追跡するのは難しい。その結果、任意の範囲での安定した実時間カメラパラメータ推定は一般に困難となる。

3章の実験では電磁気式の3次元センサを用いたが、CRTディスプレイの近くで使用するため磁場の歪みが大きく、視点の計測誤差が大きい場所が存在する。このため、重畳表示環境への適用を考えた場合、磁場の影響を受けない装置の利用を考える必要がある。そこで、本章では画像からのカメラ位置姿勢推定手法に注目し、安定してカメラパラメータを推定する手法の構築を目指す。

4.2 画像からのカメラパラメータ推定

カメラからの入力画像を現実環境としてユーザに提示するビデオシースルー型の拡張現実環境では、ユーザの視点位置とカメラの投影中心を一致させた Head Mounted Display (HMD) を用いることが多い。この場合、現実世界と仮想世界の位置合わせに必要な視点の位置姿勢の推定はカメラパラメータ推定に帰着する。そこでカメラを用いた視点位置姿勢の推定手法では、カメラパラメータを求めて視点情報とする。また、拡張現実感では実時間処理が重要となるため、用いられるカメラパラメータの推定手法にも実時間性が重視される。以下では、主に拡張現実環境を構築するための実時間カメラパラメータ推定手法を概観する。

¹通常、カメラの位置と姿勢を表すカメラパラメータは、特に外部パラメータと呼ばれるが、本論文では内部パラメータを取扱わないので、カメラパラメータの名称でカメラの位置・姿勢を表すこととする。

実環境の画像のみからカメラパラメータを推定する手法では、実環境に存在する既知の特徴点を安定して追跡することが困難であるため、電磁気式、超音波式の3次元センサと組合せて用いられることも多い。Bajuraらは3次元位置センサの誤差による位置ずれをカメラで撮影された1点のマーカの位置により補正する手法を提案した [BN95]。大島らはこの手法を拡張し、見回せる範囲を拡大した [大島 98]。Stateらは3次元位置センサの値をもとにマーカ位置を予測し、予測結果をもとにステレオカメラにより撮影される3点のマーカ位置を追跡し、カメラのパラメータを取得する手法を提案した [SLG⁺96]。このように3次元位置センサと組合せることで、安定したマーカの追跡が可能となり、正確な位置合せを安定して行えるが測定範囲が3次元位置センサの測定範囲の制限を受ける。

これに対して画像のみからカメラの位置姿勢を推定する手法として、Neumannらはカメラに撮影された既知の3点のマーカの画面座標からカメラパラメータを求めるシステムの構築例を示し、秒間1～6フレームの出力を得た [NC96]。このシステムではFischlerらの手法 [FB81] がカメラパラメータの推定アルゴリズムとして採用された。しかし、既知の3点からカメラの位置姿勢を求める問題は最大4つの解を持ち、解は一意に定まらない。Neumannらのシステムでは、4点以上の点を追跡できている場合には、選び得る3点の組合せのそれぞれから複数の解候補を求め、得られる解候補中から共通する解を選択する。また、3点のみしか追跡できていない場合には前フレームでのカメラ位置に近い解を選択する。このように、3点のみを用いる手法では、複数の解候補から解を選択する処理に工夫を要する。

一方、画像中に撮影された既知の特徴点の画面座標値からカメラの位置姿勢を推定するPnP(Perspective n-Points)問題はコンピュータビジョンの分野で古くから扱われ、多くの手法が提案されている [出口 93]。出口は文献 [出口 90] でPnP問題を一般化し、統一的な解法を導出している。この手法では既知の6点または既知の平面上の4点の画面座標値から繰り返し演算を用いずに解析的にカメラの位置姿勢を計算できる。

本章では測定範囲に制限を受けない画像のみからのカメラ位置姿勢推定手法として、

1. 一意に解を得ることができる既知の任意の 6 点を用いる手法,
2. 平面上の既知の 4 点を用いる手法,
3. 複数の解候補が求まる既知の 3 点を用いる手法

を組合せて用いる手法を実現する。必要とする特徴点の数が 6 点の場合はすべてのマーカが安定して撮影される確率は低くなるが、解析的な計算により解が一意に定まるという利点がある。平面上の既知の 4 点を用いれば解は一意に定まるが、安定して見回すことができるようにするためにはマーカの配置に工夫が必要となる。既知の 3 点から求める手法はマーカの配置は自由にできるが、解が一意に定まらない。これらの 3 つの手法を組み合わせることで各々の利点を組み合わせて利用することができると考えられる。

4.3 マーカ追跡による実時間位置合わせ

本節では入力画像からカメラの位置・姿勢を実時間で推定する提案手法の処理の手順について述べる。

4.3.1 処理の概要

提案手法において画像が入力されてから視点の位置・姿勢が取得されるまでの処理の概要は以下の通りである。

1. 入力画像から 3 次元位置が既知である複数のマーカを抽出・追跡し、画面座標を求める。
2. 抽出されたマーカの個数に応じた PnP 問題処理ルーチンでカメラパラメータを推定する。
3. 推定したカメラパラメータを元に、撮影されていないマーカの画面座標を推定する。

以降では、各段階について処理の詳細を述べる。

4.3.2 マーカの追跡

カメラパラメータを推定するために、入力画像からマーカの重心位置の画面座標を求める必要がある。本手法では、初期フレームにおいてまずフレームバッファ全体を走査して各画素の画素値を調べ、マーカとして登録されている色領域を抽出する。次に各々の色領域の重心を求め、マーカの画面座標とする。

第2フレーム以降でも、初期フレームと同様の抽出処理を繰り返すことで、マーカ領域の追跡は可能であるが、画像全体を走査する必要があるため、実時間での処理は困難である。そこで、連続するフレーム間でのマーカの画面上の移動距離は小さいと仮定をおき、前のフレームで求めたマーカ位置の重心を中心とする探索窓を設定し、探索窓領域内のみのマーカ色を抽出し、重心計算を行ない画面座標値とする。この処理を繰り返すことでマーカを追跡する。ただし、画面座標は画像中心を原点とし、水平線上の画面に向かって右方向を x 軸、鉛直線上の画面上方向を y 軸とする座標系と定める。

ここで探索窓内に探索対象となる色の画素が見つからなかった場合には、現実物体に隠蔽されたり、撮像範囲内から外れたと判断する。見つからなかったマーカの画面座標値は、以降で求めるカメラの位置姿勢情報を用いて推定され、次フレーム以降の追跡に用いられる(後述)。

追跡によって得られるマーカ領域の重心の座標値 $\mathbf{g}_i = (g_{ix}, g_{iy})^T$ は、入力画像の左下を原点とし、1画素を単位とする2次元の座標系の座標値として求められる。そこで、以下の式を用いてこの座標値を視点を原点とする3次元座標系の座標値に変換し、さらに、単位をミリメートル (mm) などの実世界で測定可能な単位に変換する。

$$\mathbf{q}_i = \begin{pmatrix} \frac{(g_{ix} - \text{scr}_x/2)\text{ccd}_x}{\text{scr}_x f} \\ \frac{(g_{iy} - \text{scr}_y/2)\text{ccd}_y}{\text{scr}_y f} \\ 1 \end{pmatrix} \quad (4)$$

ここで、 $\text{scr}_x, \text{scr}_y$ は入力画像の横、縦の画素数、 $\text{ccd}_x, \text{ccd}_y$ は撮像 CCD 面の横、縦の長さ (mm)、 f はカメラの焦点距離とする。この座標変換により、マーカの投影点の座標はカメラから投影面までの距離を1とし、カメラを原点とする3次元座標系(カメラ座標系)の座標値に変換される。

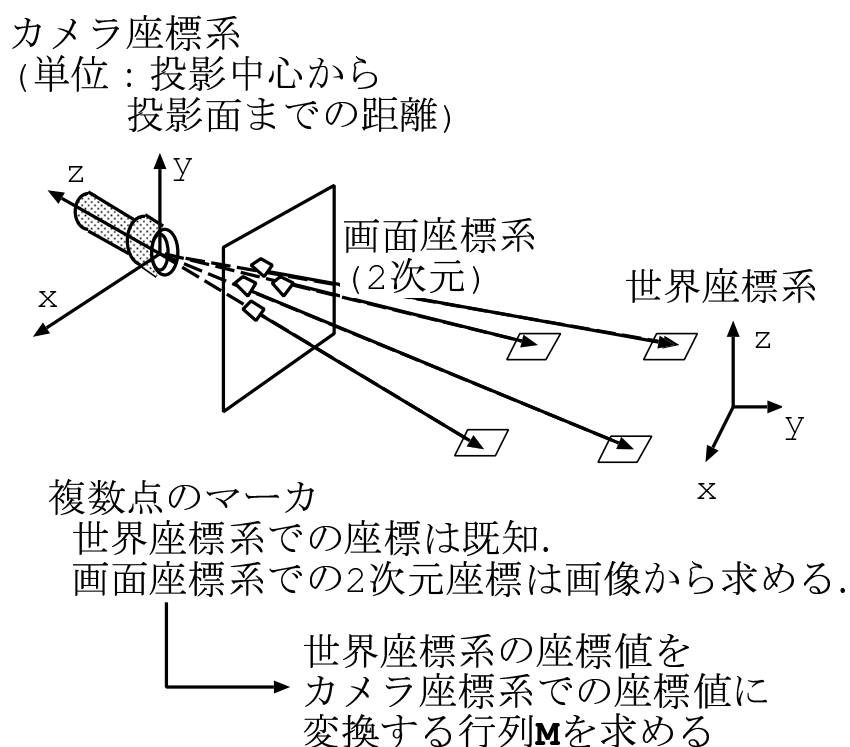


図 17 カメラパラメータ推定手法における各座標系間の関係

以上の処理から得られる複数のマーカの重心の座標値を用いてカメラパラメータを計算する手順について以降で述べる。

4.3.3 カメラパラメータの計算

カメラパラメータは実世界に設定されている基準座標系から、カメラ座標系への変換行列 (モデルビュー行列) \mathbf{M} で表すことができる。すなわち、ある点の基準座標系での座標値を \mathbf{w} , カメラ座標系での座標値を \mathbf{c} とするときに

$$\mathbf{c} = \mathbf{M}\mathbf{w} \quad (5)$$

と変換する行列である。図 17 に実世界基準座標系 (world coordinate) とカメラ座標系 (camera coordinate), モデルビュー行列 \mathbf{M} の関係を示す。

本論文では前節で画面座標を獲得したマーカの個数に応じて任意の 6 点，平面上の 4 点，任意の 3 点の画面座標値からカメラの位置姿勢を推定する手法を選択して使用する手法を提案する．その処理の概要は以下の通りである．

1. 6 点以上のマーカが抽出されていれば (2) へ，されていなければ (3) へ．
2. 6 点を用いてカメラの位置姿勢を求める．終了．
3. 平面上の 4 点のマーカが抽出されていれば (4) へ，されていなければ (5) へ．
4. 4 点を用いてカメラの位置姿勢を求める．終了．
5. 3 点以上のマーカが抽出されていれば (6) へ，されていなければ (7) へ．
6. 抽出された点のうち 3 点を用いてカメラの位置姿勢候補を求める．このとき，4 点以上の点が追跡されている場合には，すべての点をすべての位置姿勢候補で画面座標系に再投影し，画面座標の距離の和が最も小さい位置姿勢候補を出力して終了．3 点のみ抽出されている場合には，カメラの位置が前のフレームでのカメラの位置に最も近いものを選んで終了．
7. 2 点以下の場合は前のフレームでの位置姿勢をそのまま用いる．

以下ではまず，特徴点が n 点の場合に画面座標値を用いてカメラの位置姿勢を計算する一般式について述べ，その後，それぞれの手法について簡単に述べる．

n 点の場合の一般式 複数の特徴点の画面座標と基準座標の対応がわかっているとき， $\mathbf{w}_i = (w_{ix}, w_{iy}, w_{iz}, 1)^T$ ， $\mathbf{c}_i = (c_{ix}, c_{iy}, c_{iz}, 1)^T$ ， $\mathbf{s}_i = (s_{ix}, s_{iy})^T$ をそれぞれ i 番目の特徴点の基準座標系での同時座標値，カメラ座標系での同次座標値，画面座標値と定義する．透視投影の関係より， i 番目の点では次の式が成り立つ．

$$\mathbf{s}_i = \begin{pmatrix} c_{ix}/c_{iz} \\ c_{iy}/c_{iz} \end{pmatrix} \quad (6)$$

$$= \begin{pmatrix} \frac{m_{11}w_{ix} + m_{12}w_{iy} + m_{13}w_{iz} + m_{14}}{m_{31}w_{ix} + m_{32}w_{iy} + m_{33}w_{iz} + m_{34}} \\ \frac{m_{21}w_{ix} + m_{22}w_{iy} + m_{23}w_{iz} + m_{24}}{m_{31}w_{ix} + m_{32}w_{iy} + m_{33}w_{iz} + m_{34}} \end{pmatrix} \quad (7)$$

但し,

$$\mathbf{M} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (8)$$

とする. よって, これを n 個の点について連立させることで, モデルビュー行列 \mathbf{M} の成分を求めることができる [出口 90].

$$\begin{pmatrix} \mathbf{w}_1 & \mathbf{0} & -s_{1x}\mathbf{w}_1 \\ \mathbf{w}_2 & \mathbf{0} & -s_{2x}\mathbf{w}_2 \\ \vdots & \vdots & \vdots \\ \mathbf{w}_n & \mathbf{0} & -s_{nx}\mathbf{w}_n \\ \mathbf{0} & \mathbf{w}_1 & -s_{1y}\mathbf{w}_1 \\ \mathbf{0} & \mathbf{w}_2 & -s_{2y}\mathbf{w}_2 \\ \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{w}_n & -s_{ny}\mathbf{w}_n \end{pmatrix} \begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{pmatrix} = \mathbf{0} \quad (9)$$

既知の 6 点を用いる場合 方程式 (9) における左辺の $2n \times 12$ 行列のランクが 11 であるとき, 定数倍を除いて解が一意に求まる. ランクが 11 となる最小の n は 6 である. そこで $n = 6$ の場合に繰り返し演算を用いずに解析的にモデルビュー行列 \mathbf{M} を求める手順を示す.

任意の解の定数倍が一般解となるので, まず $m'_{ij} = m_{ij}/m_{34}$ として方程式 (9) を解くことを考える.

$n = 6, m'_{34} = 1$ を用いて式 (9) を変形すると,

$$\begin{pmatrix} \mathbf{w}_1 & \mathbf{0} & -s_{1x}w_{1x} & -s_{1x}w_{1y} \\ \mathbf{w}_2 & \mathbf{0} & -s_{2x}w_{2x} & -s_{2x}w_{2y} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{w}_6 & \mathbf{0} & -s_{6x}w_{6x} & -s_{6x}w_{6y} \\ \mathbf{0} & \mathbf{w}_1 & -s_{1y}w_{1x} & -s_{1y}w_{1y} \\ \mathbf{0} & \mathbf{w}_2 & -s_{2y}w_{2x} & -s_{2y}w_{2y} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{w}_6 & -s_{6y}w_{6x} & -s_{6y}w_{6y} \end{pmatrix} \mathbf{m}' = \mathbf{s} \quad (10)$$

但し,

$$\mathbf{m}' = \begin{pmatrix} m'_{11} \\ m'_{12} \\ m'_{13} \\ m'_{14} \\ m'_{21} \\ m'_{22} \\ m'_{23} \\ m'_{24} \\ m'_{31} \\ m'_{32} \\ m'_{33} \end{pmatrix}, \quad \mathbf{s} = \begin{pmatrix} s_{1x} \\ s_{2x} \\ \vdots \\ s_{6x} \\ s_{1y} \\ s_{2y} \\ \vdots \\ s_{6y} \end{pmatrix} \quad (11)$$

この変形により左辺は 12×11 の行列で表現される。そこで上部 11 行を用いて 11×11 の正方行列を作成し、逆行列を LU 分解によって求めることで、繰り返し演算を用いずに解析的に求めることができる。よって、定数時間でこの連立方程式を解くことができる。

求めたいモデルビュー行列 \mathbf{M} がカメラパラメータを表すという幾何学的な制約から、 m_{34} を定めることができる。すなわち,

$$\mathbf{m}_x = (m_{34}m'_{11}, m_{34}m'_{21}, m_{34}m'_{31})^T, \quad (12)$$

$$\mathbf{m}_y = (m_{34}m'_{12}, m_{34}m'_{22}, m_{34}m'_{32})^T, \quad (13)$$

$$\mathbf{m}_z = (m_{34}m'_{13}, m_{34}m'_{23}, m_{34}m'_{33})^T, \quad (14)$$

$$\mathbf{m}_i = (m_{34}m'_{14}, m_{34}m'_{24}, m_{34})^T. \quad (15)$$

と定義するとき, $\mathbf{m}_x, \mathbf{m}_y, \mathbf{m}_z$ がモデルビュー行列 \mathbf{M} の回転成分を表すことから得られる以下の条件式を用いる.

$$|\mathbf{m}_x| = |\mathbf{m}_y| = |\mathbf{m}_z| = 1 \quad (16)$$

$$\mathbf{m}_x \times \mathbf{m}_y = \mathbf{m}_z \quad (17)$$

このとき,

$$|\mathbf{m}_x| = \sqrt{(m_{34}m'_{11})^2 + (m_{34}m'_{21})^2 + (m_{34}m'_{31})^2} \quad (18)$$

ゆえ, 式(16)と式(18)より,

$$m_{34} = \frac{1}{\sqrt{m'_{11}{}^2 + m'_{21}{}^2 + m'_{31}{}^2}} \quad (19)$$

となって m_{34} が求まる.

既知の平面上の4点を用いる場合 平面上の4点の3次元座標を $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4$ とするとき, $\mathbf{p}_i (i = 1, 2, 3, 4)$ を $z = 0$ 平面上に移動する変換行列 \mathbf{M}_z が存在する. 例えば,

$$\mathbf{x}_a = \frac{\mathbf{p}_2 - \mathbf{p}_1}{|\mathbf{p}_2 - \mathbf{p}_1|} \quad (20)$$

$$\mathbf{y}_a = \frac{(\mathbf{p}_3 - \mathbf{p}_1) \times \mathbf{x}_a}{|(\mathbf{p}_3 - \mathbf{p}_1) \times \mathbf{x}_a|} \quad (21)$$

$$\mathbf{z}_a = \mathbf{x}_a \times \mathbf{y}_a \quad (22)$$

としたときの,

$$\begin{pmatrix} \mathbf{x}_a & \mathbf{y}_a & \mathbf{z}_a & \mathbf{p}_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} \quad (23)$$

は \mathbf{M}_z の一つである. そこで, $\mathbf{p}_i (i = 1, 2, 3, 4)$ は平面 $z = 0$ 上の4点であるとしても一般性を失わない.

$\mathbf{p}_i = (p_{ix}, p_{iy}, 0, 1)$ であるとき, \mathbf{p}_i の x, y 座標を成分にもつ 2 次元同次座標ベクトル $\mathbf{p}'_i = (p_{ix}, p_{iy}, 1)$ を定義する. このとき, 式 (9) は以下の連立方程式に変形できる.

$$\begin{pmatrix} \mathbf{p}'_1 & \mathbf{0} & -s_{1x}\mathbf{p}'_1 \\ \mathbf{p}'_2 & \mathbf{0} & -s_{2x}\mathbf{p}'_2 \\ \mathbf{p}'_3 & \mathbf{0} & -s_{3x}\mathbf{p}'_3 \\ \mathbf{p}'_4 & \mathbf{0} & -s_{4x}\mathbf{p}'_4 \\ \mathbf{0} & \mathbf{p}'_1 & -s_{1y}\mathbf{p}'_1 \\ \mathbf{0} & \mathbf{p}'_2 & -s_{2y}\mathbf{p}'_2 \\ \mathbf{0} & \mathbf{p}'_3 & -s_{3y}\mathbf{p}'_3 \\ \mathbf{0} & \mathbf{p}'_4 & -s_{4y}\mathbf{p}'_4 \end{pmatrix} \begin{pmatrix} m_{11} \\ m_{12} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{34} \end{pmatrix} = \mathbf{0} \quad (24)$$

左辺の 8×9 行列のランクが 8 のとき, 定数倍を除いて解は一意に求まる. $n = 6$ の場合と同様に $m'_{ij} = m_{ij}/m_{34}$ とおいて, 左辺が 8×8 正方行列となるように変形すると, 文献 [中沢 97] において用いられている式が導出される. 文献 [中沢 97] では焦点距離の変化に対応できるように, 焦点距離に関する 2 式を連立させ, 繰り返し演算を用いて m_{34} と焦点距離を求めている. しかし, 本手法においては繰り返し演算を用いず, 定数時間でカメラパラメータを推定するために, カメラの焦点距離は固定されているものとした. このとき, m_{34} は式 (19) で解析的に求めることができる. m_{i3} については, 式 (17) を用いて求める.

既知の 3 点を用いる場合 既知の 3 点の画面座標値が求められた場合, 最大 4 個の解が求まることが知られている. 本手法では, 計算時間が不安定な反復最適化を行わずに解析的な数値解を求める手法の中で, 比較的計算誤差が小さいとされる Finsterwalder の方法 [HLO91] を採用した. Finsterwalder の方法は既知の 3 点間の距離

$$d_1 = |\mathbf{w}_1 - \mathbf{w}_2| \quad (25)$$

$$d_2 = |\mathbf{w}_2 - \mathbf{w}_3| \quad (26)$$

$$d_3 = |\mathbf{w}_3 - \mathbf{w}_1| \quad (27)$$

と、この3点をカメラの撮像面に投影したときの画面座標系の2次元座標 \mathbf{s}_i , ($i = 1, 2, 3$) を入力とし、投影中心を原点として、撮像面水平方向を x 軸、撮像面鉛直方向を y 軸、撮像面に垂直な方向を z 軸とするカメラ座標系での3点の座標値の4つの候補 \mathbf{c}_{ij} , ($i = 1, 2, 3; j = 1, 2, 3, 4$) を出力とする。

Finsterwalder の方法でカメラ座標系での3次元座標を求めた後、カメラ座標系の3点の座標値候補 \mathbf{c}_{ij} と基準座標系の3点の座標値 \mathbf{w}_i からモデルビュー行列候補 \mathbf{M}_j を作成する必要がある。

そこでまず3点の基準座標値 \mathbf{w}_i をもとに物体座標系を定義する。

- 原点 : \mathbf{w}_1
- x 軸 : $\mathbf{w}_2 - \mathbf{w}_1$
- y 軸 : $(\mathbf{w}_3 - \mathbf{w}_1) \times (\mathbf{w}_2 - \mathbf{w}_1)$
- z 軸 : 上記で定義した x, y 軸の外積方向

物体座標系での3点の座標を \mathbf{o}_i とするとき

$$\mathbf{o}_i = \mathbf{M}_{w \rightarrow o} \mathbf{w}_i \quad (28)$$

となる変換行列 $\mathbf{M}_{w \rightarrow o}$ は以下のようにして求められる。

$$\mathbf{a}_x = \frac{\mathbf{w}_2 - \mathbf{w}_1}{|\mathbf{w}_2 - \mathbf{w}_1|} \quad (29)$$

$$\mathbf{a}_y = \frac{(\mathbf{w}_3 - \mathbf{w}_1) \times \mathbf{a}_x}{|(\mathbf{w}_3 - \mathbf{w}_1) \times \mathbf{a}_x|} \quad (30)$$

$$\mathbf{a}_z = \mathbf{a}_x \times \mathbf{a}_y \quad (31)$$

$$\mathbf{M}_{w \rightarrow o} = \begin{pmatrix} \mathbf{a}_x & \mathbf{a}_y & \mathbf{a}_z & \mathbf{w}_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} \quad (32)$$

同様に、

$$\mathbf{o}_i = \mathbf{M}_{c_j \rightarrow o} \mathbf{c}_{ji} \quad (33)$$

となる変換行列 $M_{c_j \rightarrow o}$ は

$$c_x = \frac{c_{j2} - c_{j1}}{|c_{j2} - c_{j1}|} \quad (34)$$

$$c_y = \frac{(c_{j3} - c_{j1}) \times c_x}{|(c_{j3} - c_{j1}) \times c_x|} \quad (35)$$

$$c_z = c_x \times c_y \quad (36)$$

$$M_{c_j \rightarrow o} = \begin{pmatrix} c_x & c_y & c_z & c_{j1} \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} \quad (37)$$

このとき求めるモデルビュー行列 M_j は

$$M_j = M_{c_j \rightarrow o} M_{w \rightarrow o}^{-1} \quad (38)$$

となる。

M_j から M を選択するために、4 点以上追跡できているなら、計算に用いた 3 点以外の点を再投影して、追跡点の画像平面座標と比較して最小のものを選択する。3 点のみ追跡している場合には前フレームで求めたモデルビュー行列にもっとも近いものを選択する。

観測されないマーカの位置の予測 前節までの処理により、モデルビュー行列 M が求まる。しかし、視点・視線の自由な変更を可能にし、カメラの位置と姿勢を安定に推定するためには、広範囲に配置されたマーカを順次切り替えて利用する必要がある。このため、各フレームで抽出・追跡に失敗したマーカについてもモデルビュー行列 M を用いてカメラ座標系への変換を行ない、画面座標値を推定する。この座標値は画像枠外になることもあり得る。この処理により、実物体によって隠蔽されたり、撮像範囲外に存在するため追跡できないマーカの位置を推定しておき、再び見えるようになったときや、撮像範囲内に新しく入って来た場合に追跡を再開もしくは開始することができる。

4.4 ビデオスルー型拡張現実感試作システムの構築

前節で提案した実時間カメラパラメータ推定手法の動作を確認するために、ビデオスルー型の拡張現実環境提示システムを試作した。

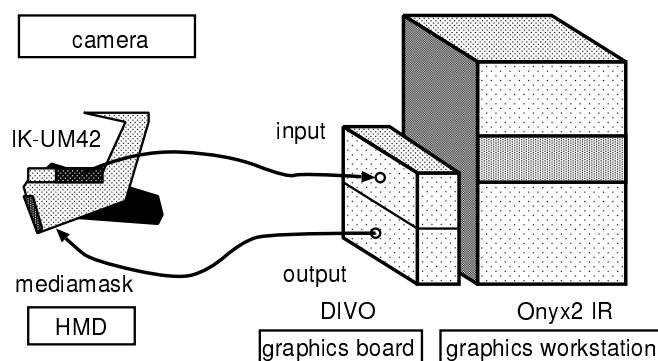


図 18 ビデオシースルー型拡張現実感試作システムのシステム構成図

4.4.1 プロトタイプシステムの構築

システムの実装には動画像の入出力および動画像のテクスチャマッピングが可能な汎用グラフィックワークステーション (SGI Onyx2 IR + DIVO)、HMD(オリンパス media mask)、小型 CCD カメラ (東芝 IK-UM42) を用いた。システムの構成を図 18 に示す。試作システムでは、まず HMD に取り付けられた実環境撮影用カメラからの画像を入力とし、提案手法によりカメラ位置姿勢を推定する。次に、得られたカメラ位置姿勢を用いて仮想環境をレンダリングし、カメラ画像に重畳して HMD に提示する。

図 19 では試作システムにおいて画像が入力されてから合成画像が提示されるまでの処理の概要を示す。図 19 に示す手順は以下の通りである。

1. 予め求めておいた歪み補正パラメータを用いて提示用入力画像を近似的に補正する。
2. 提案手法によりカメラパラメータを推定する。
3. 出力画像の背景に入力画像を描画しておき、得られたカメラパラメータを用いて仮想環境をレンダリングする。1. へ戻る。

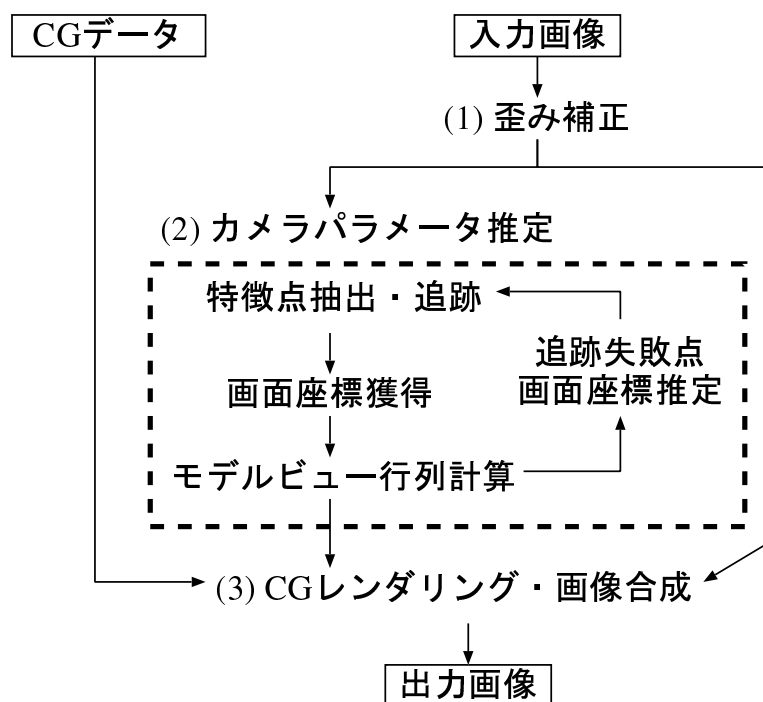


図 19 ビデオシースルー型拡張現実感試作システムの処理の概要

4.4.2 入力画像の実時間歪み補正

現実環境を撮影するカメラは理想的なピンホールカメラではないため、入力画像には樽型の歪みが存在する。このため、正確なカメラパラメータを推定するには、視点の位置・姿勢を求めるために使用する既知の点の画面座標値に対して歪みの逆変換計算が必要となる。

この補正を行なうには

- 入力画像から既知の特徴点の画面座標値を取得してから画面座標値を補正する、
- 歪みを補正した画像を作成してから画面座標値を取得する

という2つの方法が考えられる。

画面座標値を補正する方法は計算量も少なく実現が容易であるが、ビデオシー
スルー型の拡張現実環境を作成した場合に補正されていない画像に対して補正さ
れた画面座標値を用いて仮想環境を合成するとずれが生じる。歪みを補正した画
像を作成してから画面座標値を取得する方法ではこのようなずれは生じないが、
汎用計算機での実時間処理が困難となる。

そこで試作システムは入力画像の歪みを、グラフィックスボードによって実現
されている動画像テクスチャマッピング機能を用いて近似的に補正した。歪みの
モデルには Tsai のモデルを採用した [Tsa87]。補正手順は以下のとおりである。

1. Tsai のカメラキャリブレーション法により、カメラの歪み係数 κ_1 を求めて
おく。
2. 入力画像について中心を通る水平線と鉛直線および対角線による放射線上
の全ての画素について、補正後に移動する座標を求める。画像中心を原点
とした場合、補正後の座標 \mathbf{q} を補正前の座標 \mathbf{p} を用いて以下の式で求める。

$$\mathbf{q} = \mathbf{p}(1 + \kappa_1 r^2)$$

3. 以上の操作によるずれ $\mathbf{q} - \mathbf{p}$ が変化する点を頂点とする蜘蛛の巣状の格子
パターンを設定し、格子点状の画素が補正後の画素に対応するようにテク
スチャマッピングする。

入力動画像のテクスチャマッピングはハードウェアによって行なわれるため、
上記の格子パターンを一度計算しておけば、補正画像は実時間で提示可能となる。
以上の処理により樽型歪みを近似補正した画像を得ることができる。この画像を
入力として提案手法によるカメラパラメータの推定と、合成画像の作成を行う。

4.4.3 実画像と仮想物体の合成

仮想物体のレンダリングに用いる投影パラメータとして、CCD カメラの撮像
面の大きさや焦点距離を設定する。次に、仮想物体のレンダリングに必要となる
カメラ情報として提案手法で求めたモデルビュー行列 \mathbf{M} を用いる。以上のパラ
メータを用いて世界座標系で仮想物体を描画することで、実世界画像と仮想物体
を位置ずれなく合成することができる。

4.4.4 試作システムの実行結果

試作システムによって CG を合成した複合現実環境の様子を図 20, 21に示す. 図 20において酒樽は仮想物体である. 図 20(3) ではマーカの一つが実物体により隠蔽されているが残りの 3 点により追跡され, 図 20(4) で再び追跡を開始している. 図 21において窓, 窓の外の風景, 棚やティーポットは仮想物体である. 図 21(1) に初期フレームを示す. 柱に設置された 6 点のマーカからカメラの位置と姿勢を推定している. 図 21(2) では, 初期フレームでカメラ位置の推定に用いられたマーカはユーザの見回しにより, 撮像範囲外に移動している. しかし, 初期フレームで撮像範囲外にあったマーカが新たに追跡され, カメラ位置姿勢の推定に用いられている. 図 21(3) では撮像範囲内にある平面上の 4 点のマーカのみからカメラの位置と姿勢を推定している.

試作システムの画像更新レートは, 実行例に示す程度の仮想物体を合成する場合で 15 フレーム/秒である. ビデオシースルー型拡張現実環境の原理上, 実環境と仮想環境との時間的なずれは発生しない. その反面, 実環境が撮影されてから拡張現実環境がユーザに提示されるまでに, 画像取り込み, 合成処理, 画像出力でそれぞれ 1 フレームの遅延が発生し, 合計 3 フレーム時間 (約 0.1 秒) 遅延する.

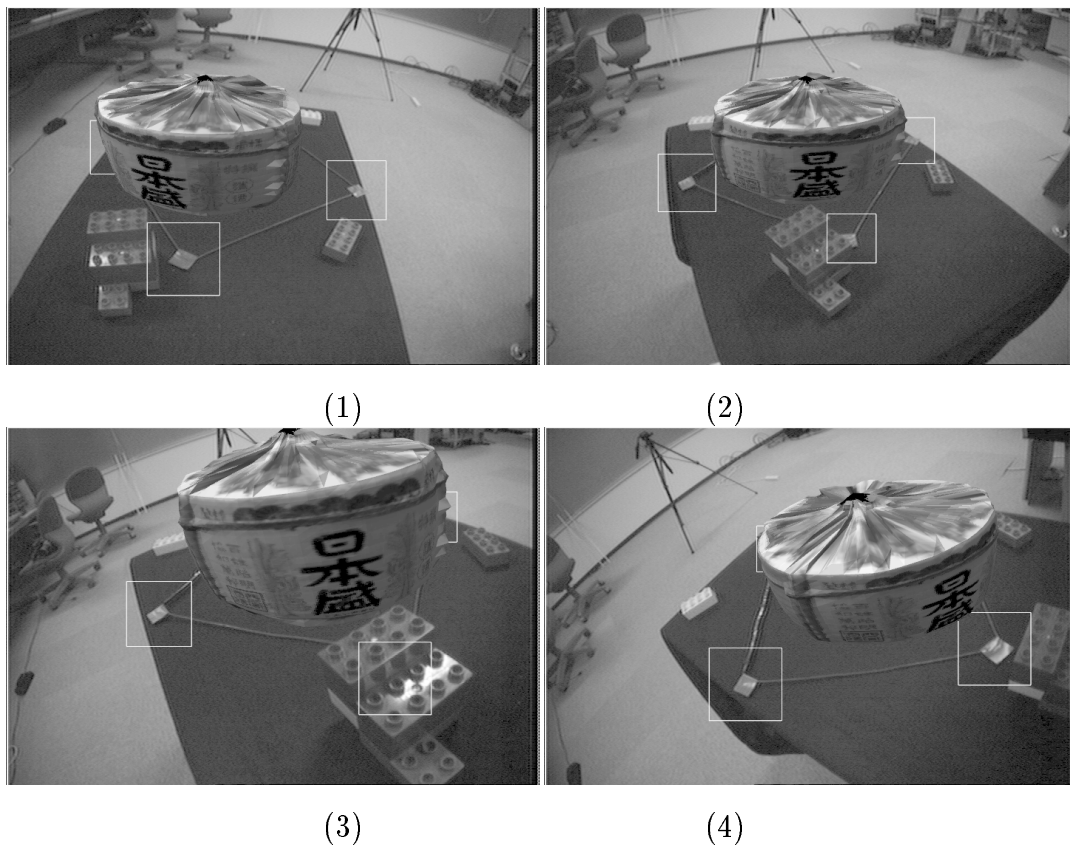


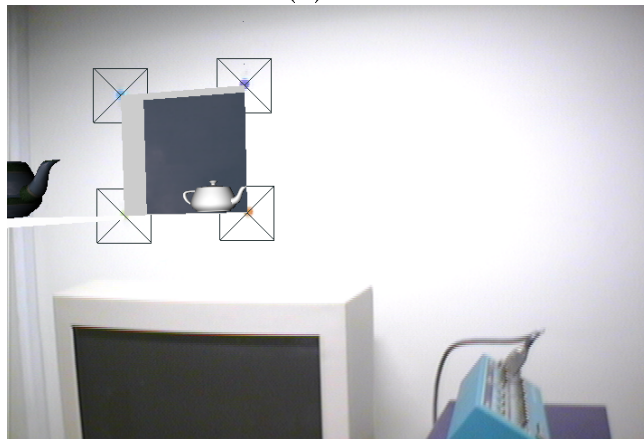
図 20 ビデオシースルー型拡張現実感試作システムの実行例 (その 1)



(1)



(2)



(3)

図 21 ビデオシースルー型拡張現実感試作システムの実行例 (その 2)

4.5 考察

本手法によりカメラ画像から得られる特徴点(マーカ)の個数によってカメラ位置を推定する手法を切替えることにより実時間で解析的にカメラパラメータを推定し、正確な位置合わせを実現することができる。また、入力画像内のマーカでカメラパラメータを推定できた場合、入力画像の外側に存在するマーカや他の物体によって一時的に隠されたマーカの位置を推定しておいて、新たに入力画面内に観測された場合の追跡を可能にする。この結果、常に視野領域内に3点以上のマーカが撮影される間隔で識別可能なマーカを配置すれば、広範囲で位置姿勢を推定できる。また、提案した手法の動作を確認するためにビデオシースルー型拡張現実感システムを試作し、その動作例を示すことで提案手法の有効性を確認した。

しかしながら、実際に情報作業空間として適用するためには

1. 1つのカメラからの画像を両眼に提示するので奥行き感覚が掴みにくい、
2. マーカ以外の実物体の奥行き情報は画像からは得られていないので、仮想物体より手前にモデル化されていない実物体が存在する場合に、矛盾した合成画像になる。

などの問題点がある。1.の問題に関しては左右両眼用の画像を撮影し、それぞれにCGを合成して提示することで解決できる。2.の問題を解決するには、実環境の奥行き情報を実時間で取得して、仮想物体と実物体を矛盾なく重ねる必要がある。これには、ステレオ画像から視差を求め、実環境の奥行き情報を獲得してCGオブジェクトを合成する手法による解決などが考えられる[神原98, KYO+96]。

また、現在の実装上の問題として画像更新レートが不十分であるため、ユーザが見回す速度が速くなるとマーカのトラッキングに失敗することがある。将来の計算機の速度の向上などによりある程度改善されると考えられるが、ユーザの視線位置や方向などをカルマンフィルタ等を用いて予測する手法[WO95]の導入によりさらなる改善が期待できる。

4.6 結言

本章では，拡張現実環境における位置ずれ問題を解決するために，従来コンピュータビジョンの分野で研究されてきた P6P，P4P 問題の解法と Finsterwalder による P3P 問題の解法を組み合わせる手法を拡張現実環境の位置ずれ問題に適用した．この手法は画面から得られる特徴点の個数によってカメラ位置を推定する手法を切替えることにより実時間で解析的にカメラ位置を推定し，正確な位置合わせを実現することができる．また，提案した手法の動作を確認するためにビデオシースルー型拡張現実感システムを試作し，その動作例を示した．

3章で述べた重畳表示環境は光学的シースルーの利点を利用する物であり，本章で扱ったビデオシースルー型拡張現実をそのまま適用することはできない．しかし，ユーザの視点とカメラ位置をハーフミラー等を用いて正確に重ね，ユーザの視点位置からの画像を計算機に入力可能な HMD が近い将来登場する可能性も高い．このような HMD を用いることで本章で述べた手法は重畳表示環境へも適用可能になると考えられる．

5. 結論

本論文では、情報のブラウジング時にブラウジング対象となる情報の全体構造と注目部分の詳細を同時に提示できる情報作業空間の構築を目標とし、3次元ユーザインタフェースと複合現実感を用いて大規模情報を効率良くブラウジングする手法を実現した。具体的には、3次元ユーザインタフェースによる順序付き階層構造情報の視覚化手法、CRTとHMDを併用する拡張現実環境としての重畳表示環境による情報作業空間の構築、拡張現実環境による空間的ずれを検討するためのビデオシースルー型拡張現実環境の構築法についてそれぞれ検討した。

2章では大規模な情報を利用するための順序つき階層構造情報の表示手法としては従来のCone Treeの改良型であるSpiral Treeを提案した。提案手法Spiral Treeが順序を利用する検索に有効であることを確認するために、ノード探索タスクを用いた比較実験を行った。実験の結果、Spiral Treeを用いた場合の方がCone Treeを用いる場合に比べてタスク達成時間や、タスク達成に必要な作業数を減少することが分かった。これはSpiral TreeがCone Treeに比べて注目階層の一覧性が高く、階層内の順序を利用することができるという特徴によると思われる。また、Spiral Treeで実際に階層内順序が有効に利用されているかを確認するための実験も行った。階層内の順序を利用できるSpiral Treeと利用できないSpiral Treeを用いてのノード探索タスクを用いた比較実験の結果、順序が利用できる場合の方がタスク達成時間などが減少する傾向が観察され、Spiral Treeでは順序が利用されている可能性が高いことを示した。

3章では、大規模情報を利用するための仮想環境の提示手法として、CRTによる提示領域を透過型HMDを用いた没入型仮想環境を用いて拡大する重畳表示環境を提案した。これにより文書などの情報の検討や編集を行うための高い解像度と、多くの情報を配置できる広い仮想環境を同時に利用することができると考えられる。この重畳表示環境を評価するために2つの実験を実施した。1つ目の実験ではターゲットを探索する作業をCRTのみ用いる場合、HMDのみ用いる場合、重畳表示環境を用いる場合の3通りでおこなって比較した。その結果、粗い解像度で十分な大きなターゲットを探索する場合には、HMDを用いることで素早くターゲットを探索することができ、ターゲットが小さい場合には、CRTを利

用する方が探索時間を短縮できることが実験的に示された。提案してきた重畳表示環境ではその両方の利点を利用することにより、どちらの場合でも短い時間で探索することができていることが確認された。また、2つ目の実験の結果、2つの領域の境界に起こるいくつかのずれが違和感を与え物体配置の作業の効率に与える影響を確認する実験を行なった。この実験には重畳表示環境を実際に使用する場合を想定した、CRTによる表示領域とHMDによる表示領域の間で仮想物体を移動させるタスクを用いた。この結果、2つの領域間の境界では立体を正しく認識するのが困難になって仮想物体の配置に時間がかかることや、立体視ができないCRTによる表示領域では仮想物体の奥行きが分かりにくくなって仮想物体の配置に時間がかかるという問題点を確認した。

4章では、拡張現実感における位置ずれ問題を解決するため、実環境を撮影した画像から、カメラの位置・姿勢を実時間で推定する手法について検討し、画像内に撮影される既知の6点、既知の平面上の4点、既知の3点を用いてカメラの位置・姿勢を実時間で推定する手法を提案した。また、提案したカメラ位置姿勢推定手法を用いるビデオシースルー型拡張現実感システムを試作し、提案手法の有効性を示した。

以上、2, 3, 4各章において提案し実装した手法を組み合わせることで、計算機で管理される階層構造をブラウジングするための効果的な情報作業空間を提供することができると考えられる。即ち、2章で提案したSpiral Treeによって視覚化した階層構造情報を、3章で提案した重畳表示環境内に提示する事で各ノード上に書かれたノード名を読むのに十分な解像度を保持したまま、1000ノードを越えるような大規模情報を提示可能になる。この時のCRT提示領域とHMD提示領域に発生する位置ずれは、視点位置に小型カメラを取り付け、取り付けたカメラからの画像を用いて4章で提案したカメラ位置姿勢推定手法の適用により軽減させることができると考えられる。

本論文でブラウジングの対象とした情報は順序付きの階層構造情報であった。階層構造は情報の管理などにも良く使用され、重要な構造であると考えられるが、ブラウジングの対象となる全ての情報が階層構造で表現される訳ではない。より汎用的な情報のブラウジング手法を提供するためには一般のグラフ構造の3次元

視覚化手法についても検討が必要であると考えられる。また、現在までに構築したシステムはユーザが同時に使用できるデバイスに関する制限などから、3次元拡張現実環境全体を情報作業空間として利用するには至らなかった。しかしながらこれらのデバイスに関する問題は近い将来、解決されるものと考えられる。実際にHMDの小型化などに伴って液晶シャッター眼鏡とHMDの併用が可能になった場合の全領域が両眼立体視可能な重畳表示環境の利用、ハーフミラー等を利用してユーザの視点とカメラの視点を仮想的に一致させた透過型HMDを用いた場合に4章で述べた手法による重畳表示環境の位置ずれ問題の解決等を実現していくことで、本論文で提案した情報作業環境は、より効果的に情報をブラウジングするユーザインタフェースの提供を可能になると考えられる。

謝辞

本研究を行う機会を与えて頂き、研究の全過程を通して直接懇切なる御指導、御鞭撻を賜ったソフトウェア基礎講座 横矢 直和 教授 に衷心より感謝の意を表します。

本研究の遂行にあたり、終始有益な御助言と励ましの言葉を頂いた像情報処理学講座 千原 國宏 教授 に厚く御礼申し上げます。

本研究の全過程を通して、様々な相談に応じて頂き、懇切丁寧なる御指導、御鞭撻を賜ったソフトウェア基礎講座 竹村 治雄 助教授 に心より厚く御礼申し上げます。

研究初期に様々な御助言を頂いた大阪大学基礎工学部情報工学科 萩原 兼一 教授 に感謝致します。

本研究の遂行にあたり、適宜的確な御助言を下された像情報処理学講座佐藤 宏介 助教授に感謝致します。

本研究を通じて、終始熱心に相談に応じて頂き、有益な御助言を頂いたソフトウェア基礎講座 岩佐 英彦 助手 に感謝致します。

研究の相談に応じて頂き、様々な御助言を頂いた情報科学センター 片山 喜章 助手 およびソフトウェア基礎講座 山澤 一誠 助手 に心より御礼申し上げます。

学部時代からの親友として、本学入学以来常に苦楽を共にし、研究のみならず公私に渡る様々な問題の相談相手となって頂いたソフトウェア基礎講座 清川 清君 に心より感謝の意を表します。

本研究を遂行する期間に、公私に渡る多くの御助言と御協力を頂き、苦楽を共有した元ソフトウェア基礎講座事務補佐員 村上 和代 さんに心より感謝致します。

また、物心両面において常に温かい御支援を頂き、また本研究の評価実験においてもこころよく御協力を頂いたソフトウェア基礎講座の諸氏、OB、ならびに、ソフトウェア基礎講座事務補佐員 福永 博美 女史に感謝致します。

最後に、私を常に見守り多くの面で支えて頂いた両親と弟に深く感謝します。

参考文献

- [AS94] C. Ahlberg and B. Shneiderman, "Visual information seeking: Tight coupling of dynamic query filters with starfield displays," *Proc. CHI'94: Human Factors in Computing Systems*, pp.313-317. ACM, 1994.
- [Azu97] R. T. Azuma, "A survey of augmented reality," *Presence*, 6, 4, pp.355-385, 1997.
- [BN95] M. Bajura and U. Neumann, "Dynamic registration correction in video-based augmented reality systems," *IEEE Computer Graphics and Applications*, 15, 5, pp.52-60, September 1995.
- [CH87] S. K. Card and A. H. Jr. Hendedrson, "A multiple, virtual workspace interface to support user task switching," *Proc. of the CHI+GI 1987*, pp.53-59, 1987.
- [CK95] J. Carriere and R. Kazman, "Interacting with huge hierarchies: Beyond cone trees," *Proc. IEEE Sympo. on Information Visualization*, pp.74-81, 1995.
- [CNSD93] C. Cruz-Neira, J. D. Sandin, and T. A. Defanti, "Surround-screen projection-based virtual reality: The design and implementation of the cave," *Proc. of SIGGRAPH'93*, pp.135-142, 1993.
- [CRM91] S. K. Card, G. G. Robertson, and J. D. Mackinlay, "The Information Visualizer: An information workspace," *Proc. CHI'91: Human Factors in Computing Systems*, pp.181-188, 1991.
- [FB81] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, 24, 6, pp.381-395, 1981.

- [FMHS93] S. Feiner, B. MacIntyre, M. Haupt, and E. Solomon, "Windows on the world: 2d windows for 3d augmented reality," *Proc. of the ACM Symposium on User Interface Software and Technology (UIST'93)*, pp.145-155, 1993.
- [FMS93] S. Feiner, B. MacIntyre, and D. Seligmann, "Knowledge-based augmented reality," *Communications of the ACM*, 36, 7, pp.53-61, 1993.
- [FS91] S. Feiner and A. Shamash, "Hybrid user interfaces: Breeding virtually bigger interfaces for physically smaller computers," *Proc. of the ACM Symposium on User Interface Software and Technology (UIST'91)*, pp.9-17, 1991.
- [Fur86] G. W. Furnas, "Generalized fisheye views," *Proc. CHI'86: Human Factors in Computing Systems*, pp.16-23. ACM, 1986.
- [GAB⁺98] S. Gibbs, C. Arapis, C. Breiteneder, V. Lalioti, S. Mostafawy, and J. Speier, "Virtual studios," *IEEE Multimedia*, 5, 1, pp.18-35, 1998.
- [HLO91] R. M. Haralick, C.-N. Lee, and K. Ottenberg, "Analysis and solutions of the three point perspective pose estimation problem," *Proc. CVPR'91*, pp.592-598, 1991.
- [JR94] L. John and R. Ramana, "Laying out and visualizing large trees using a hyperbolic space," *Proc. of the ACM Symposium on User Interface Software and Technology (UIST'94)*, pp.13-14, 1994.
- [JS91] B. Johnson and B. Shneiderman, "Treemaps: A space-filing approach to the visualization of hierarchical information structures," *Proc. of Visualization'91*, pp.284-291, 1991.
- [KV98] K. N. Kutulakos and J. R. Vallino, "Calibration-free augmented reality," *IEEE Trans. Visualization and Computer Graphics*, 4, 1, pp.1-20, 1998.

- [KYO⁺96] T. Kanade, A. Yoshida, K. Oda, H. Kano, and M. Tanaka, "Stereo machine for video-rate dense depth mapping and its new application," *Proc. CVPR'96*, pp.196-202, 1996.
- [MRC91] J. D. Mackinlay, G. G. Robertson, and S. K. Card, "The perspective wall: Detail and context smoothly integrated," *Proc. CHI'91: Human Factors in Computing Systems*, pp.173-179, 1991.
- [MRC95] J. D. Mackinlay, R. Rao, and S. K. Card, "An organic user interface for searching citation links," *Proc. CHI'95: Human Factors in Computing Systems*, pp.67-73, 1995.
- [NC96] U. Neumann and Y. Cho, "A self-tracking augmented reality system," *Proc. VRST 96*, pp.109-115, 1996.
- [OT96] T. Oishi and S. Tachi, "Methods to calibrate projection transformation parameters for see-through head-mounted displays," *Presence*, 5, 1, pp.129-142, 1996.
- [OTK⁺96] T. Okuma, H. Takemura, Y. Katayama, H. Iwasa, and N. Yokoya, "3d visualization of hierarchical ordered information," *Advances in Applied Ergonomics(Proc. ICAE'96)*, pp.668-671, 1996.
- [R. A86] R. A. Bolt, 鶴岡雄二 (訳), マン-マシン・インターフェース進化論, パーソナルメディア, 1986.
- [RCM93] G. G. Robertson, S. K. Card, and J. D. Mackinlay, "Information visualization using 3D interactive animation," *Commun. the ACM*, 36, 4, pp.57-71, April 1993.
- [Ren94] E. Rennison, "Galaxy of news an approach to visualizing and understanding expansive news," *Proc. of the ACM Symposium on User Interface Software and Technology (UIST'94)*, pp.3-12. ACM, November 1994.

- [RMC91] G. G. Robertson, J. D. Mackinlay, and S. K. Card, "Cone Trees: Animated Visualizations of Hierarchical Information," *Proc. CHI'91: Human Factors in Computing Systems*, pp.189-194, 1991.
- [RN95] J. Rekimoto and K. Nagao, "The world through the computer: Computer augmented interaction with real world environments," *Proc. of the ACM Symposium on User Interface Software and Technology (UIST'95)*, pp.29-36, 1995.
- [RPh⁺95] R. Rao, J. O. Pedersen, M. A. Hearst, J. D. Mackinlay, S. K. Card, L. Masinter, P-K. Halvorsen, and G. G. Robertson, "Rich Interaction in the Digital Library," *Commun. the ACM*, 38, 4, pp.29-39, April 1995.
- [SLG⁺96] A. State, M. A. Livingston, W. F. Garrett, G. Hirota, M. C. Whitton, E. D. Pisano, and H. Fuchs, "Technologies for augmented reality systems: Realizing ultrasound-guided needle biopsies," *Proc. SIGGRAPH 96*, pp.439-446, 1996.
- [SMR⁺97] T. Starner, S. Mann, B. Rhodes, J. Levine, D. Kirsch, R. W. Picard, and A. Pentland, "Augmented reality through wearable computing," *Presence*, 6, 4, pp.386-398, 1997.
- [TGW⁺95] M. Tuceryan, D. S. Greer, R. T. Whitaker, D. E. Breen, C. Crampton, E. Rose, and K. H. Ahlers, "Calibration requirements and procedures for monitor-based augmented reality," *IEEE Trans. Visualization and Computer Graphics*, 1, 3, pp.255-273, 1995.
- [Tsa87] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal of Robotics and Automation*, RA-3, 4, pp.323-344, 1987.

- [Wel93] P. Wellner, "Interacting with paper on the digitaldesk," *Communications of the ACM*, 36, 7, pp.87-96, 1993.
- [WO95] J.-R. Wu and M. Ouhyoung, "A 3d tracking experiment on latency and its compensation methods in virtual environments," *Proc. of the ACM Symposium on User Interface Software and Technology (UIST'95)*, pp.29-36, 1995.
- [寺岡 96] 寺岡照彦, 丸山稔, "ユーザの視点に基づく適応型 3 次元インタフェース", 電子情報通信学会技術報告, pp.MVE96-52, 1996.
- [出口 90] 出口光一郎, "射影幾何学による PnP カメラ補正問題の統一的解法", コンピュータビジョン'90 シンポジウム, pp.41-50, 1990.
- [出口 93] 出口光一郎, "カメラキャリブレーション手法の最近の動向", 情処研報 93-CV-82-1, pp.1-8, 1993.
- [小池 94] 小池英樹, 吉原大敬, "対話型システムにおける大規模階層構造視覚化へのフラクタルの応用", 情報処理学会論文誌, 35, 12, pp.2703-2711, 1994.
- [神原 98] 神原誠之, 大隈隆史, 竹村治雄, 横矢直和, "複合現実感のための実時間ステレオ画像合成", 映像情報メディア学会技術報告ヒューマンインフォメーション, pp.Vol.22, No.33, 31-36, 1998.
- [水野 93] 水野浩孝, 青木由起子, 辻洋, "データ可視化技法を用いた情報検索方式の提案", 第 9 回ヒューマンインタフェースシンポジウム論文集, pp.79-82, 1993.
- [大隈 95] 大隈隆史, 竹村治雄, 片山喜章, 横矢直和, "三次元視覚化における奥行き提示方法の効果について", 1995 年電子情報通信学会総合大会講演論文集, pp.A-264, 1995.

- [大隈 97] 大隈隆史, 清川清, 竹村治雄, 横矢直和, “ビデオシースルー型拡張現実のための実画像からのカメラパラメータの実時間推定”, 電子情報通信学会技術報告, pp.PRMU97-113, 1997.
- [大隈 98] 大隈隆史, 竹村治雄, 岩佐英彦, 片山喜章, 横矢直和, “Spiral tree: 順序付き階層構造の 3 次元視覚化手法”, 電子情報通信学会論文誌 A, J81-A, 2, pp.280-288, February 1998.
- [大島 98] 大島登志一, 佐藤清秀, 山本裕之, 田村秀行, “AR² ホッケー: 協調型複合現実感システムの実現”, 日本バーチャルリアリティー学会論文誌, 3, 2, pp.55-60, 1998.
- [中沢 97] 中沢裕二, 中野誠巳, 小松隆, 斎藤隆弘, “画像中の特徴点に基づく実写画像と CG 画像との動画像合成システム”, 映像情報メディア学会誌, 51, 7, pp.1086-1095, 1997.

研究業績

学術論文誌

1. 大隈 隆史, 竹村 治雄, 岩佐 英彦, 片山 喜章, 横矢 直和: “Spiral Tree: 順序付き階層構造の3次元視覚化手法”, 電子情報通信学会論文誌 A, Vol.J81-A, No. 2, pp.280-288, 1998. (本論文2章に関連)

国際会議

1. Okuma, T., Takemura, H., Katayama, Y., Iwasa, H. and Yokoya, N. : “3D visualization of hierarchical ordered information : Spiral Trees ,” *Advances in Applied Ergonomics(Proc. ICAE'96)*, pp.668-671, Istanbul, Turkey, May 1996. (本論文2章に関連)
2. Okuma, T., Kiyokawa, K., Takemura, H. and Yokoya, N. : “An augmented reality system using a real-time vision based registration,” *Proc. 14th IAPR Int. Conf. on Pattern Recognition (ICPR'98)*, pp.1226-1229, August 1998. (本論文4章に関連)
3. Yokoya, N., Takemura, H., Okuma, T. and Kanbara, M. : “Stereo vision based video see-through mixed reality”, *Proc. 1st Int. Sympo. on Mixed Reality (ISMR'99)*, Yokohama, Japan, March 1999 (発表予定).
4. Kanbara, M., Okuma, T., Takemura, H. and Yokoya, N. : “Real-time composition of stereo images for video see-through augmented reality”, *Proc. 6th IEEE Int. Conf. on Multimedia Computing and Systems (ICMCS'99)*, Firenze, Italy, June 1999 (発表予定).

研究会・シンポジウム

1. 大隈 隆史, 竹村 治雄, 片山 喜章, 岩佐 英彦, 横矢 直和: “Spiral Tree: 順序つき階層構造情報の三次元視覚化”, 計測自動制御学会 第11回 ヒューマンインタフェースシンポジウム論文集, pp.369-374, October 1995. (本論文2章に関連)

2. 大隈 隆史, 竹村 治雄, 片山 喜章, 岩佐 英彦, 横矢 直和: “順序付き階層構造情報の三次元視覚化の一手法”, 情報処理学会研究報告, HI63-3, November 1995. (本論文 2章に関連)
3. 土本 光一, 大隈 隆史, 竹村 治雄, 片山 喜章, 岩佐 英彦, 横矢 直和: “三次元視覚化における奥行き提示方法の効果に関する実験的評価”, 電子情報通信学会技術報告, IE95-130, February 1996.
4. 大隈 隆史, 竹村 治雄, 片山 喜章, 岩佐 英彦, 横矢 直和: “重畳表示環境による作業領域拡大の実験評価”, 電子情報通信学会技術報告, IE96-109, February 1997. (本論文 3章に関連)
5. 大隈 隆史, 清川 清, 竹村 治雄, 横矢 直和: “ビデオシースルー型拡張現実のための実画像からのカメラパラメータの実時間推定”, 電子情報通信学会技術報告, PRMU97-113, October 1997. (本論文 4章に関連)
6. 仲村 元亨, 大隈 隆史, 竹村 治雄, 横矢 直和: “AR 環境におけるカード型情報管理システム”, 第 13 回ヒューマンインタフェースシンポジウム論文集, pp.353-358, October 1997.
7. 仲村 元亨, 大隈 隆史, 竹村 治雄, 横矢 直和: “AR 環境における情報の生成・管理手法の一考察”, 情報処理学会研究報告, HI76-6, January 1998.
8. 世利 至彦, 大隈 隆史, 清川 清, 竹村 治雄, 横矢 直和: “汎用 3 次元ユーザインタフェースツールキットの実装”, 電子情報通信学会技術報告, IE97-157, February 1998.
9. 神原 誠之, 大隈 隆史, 竹村 治雄, 横矢 直和: “複合現実感のための実時間ステレオ画像合成”, 映像情報メディア学会技術報告, Vol.22, No.33, pp.31-36, June 1998.
10. 大隈 隆史, 神原 誠之, 竹村 治雄, 横矢 直和: “実時間位置合わせによるビデオシースルー型複合現実”, 日本バーチャルリアリティ学会 複合現実感技術研究会, September 1998. (本論文 4章に関連)

全国大会

1. 大隈 隆史, 竹村 治雄, 片山 喜章, 横矢 直和: “三次元視覚化における奥行き提示方法の効果について”, 1995 年電子情報通信学会春季総合大会講演論文集, A-264, March 1995.
2. 大隈 隆史, 竹村 治雄, 片山 喜章, 岩佐 英彦, 横矢 直和: “順序つき階層構造の三次元視覚化手法の提案と評価”, 第 51 回 (平成 7 年後期) 情報処理学会全国大会講演論文集, 7U-3, September 1995. (本論文 2 章に関連)
3. 大隈 隆史, 竹村 治雄, 岩佐 英彦, 片山 喜章, 横矢 直和: “三次元視覚化手法 Spiral Tree による情報のブラウジング支援”, システム制御情報学会研究発表講演会, 2011, May 1996. (本論文 2 章に関連)
4. 大隈 隆史, 竹村 治雄, 片山 喜章, 岩佐 英彦, 横矢 直和: “透過型 HMD を用いた重畳表示による三次元ユーザインタフェース”, 第 53 回 (平成 8 年後期) 情報処理学会全国大会講演論文集, 4H-2, September 1996. (本論文 3 章に関連)
5. 大隈 隆史, 竹村 治雄, 岩佐 英彦, 片山 喜章, 横矢 直和: “三次元視覚化手法 Spiral Tree による仮想空間内での情報ブラウジング”, 第 39 回自動制御連合講演会, 3035, October 1996. (本論文 2 章に関連)
6. 仲村 元亨, 大隈 隆史, 竹村 治雄, 横矢 直和: “小型カメラを用いた AR 空間における仮想付箋の試作”, 1997 年電子情報通信学会総合大会講演論文集, A-16-17, March 1997.
7. 世利 至彦, 大隈 隆史, 竹村 治雄, 横矢 直和: “汎用 3 次元ユーザインタフェースツールキットの試作”, 情報処理学会第 55 回 (平成 9 年後期) 全国大会講演論文集, 1AA-1, September 1997.
8. 大隈 隆史, 竹村 治雄, 岩佐 英彦, 片山 喜章, 横矢 直和: “透過型 HMD と CRT の併用による重畳表示環境の有効性の一検討”, 情報処理学会第 55 回

(平成9年後期) 全国大会講演論文集, 1AA-6, September 1997. (本論文3章に関連)

9. 仲村 元亨, 大隈 隆史, 竹村 治雄, 横矢 直和: “AR 環境を用いたカード型情報管理システムの試作”, 情報処理学会第55回(平成9年後期) 全国大会講演論文集, 1AA-7, September 1997.
10. 神原 誠之, 大隈 隆史, 竹村 治雄, 横矢 直和: “実時間ステレオ画像合成を用いた複合現実感システムの試作”, 1998年電子情報通信学会情報・システムソサイエティ大会講演論文集, No.D-12-74, September 1998.

解説記事

1. 竹村 治雄, 清川 清, 大隈 隆史, “仮想環境の共有技術(ネットワーク)”, 日本バーチャルリアリティ学会講習会資料, 1/20, 1998.