

順序付き階層構造情報の三次元視覚化の一手法

大隈隆史 竹村治雄 片山喜章 岩佐英彦 横矢直和

奈良先端科学技術大学院大学

〒630-01 奈良県生駒市高山町 8916-5

{takasi-o, takemura, katayama, iwasa, yokoya}@is.aist-nara.ac.jp

大規模な情報を利用する際には閲覧をいかに効率良く行なうかが重要になる。本稿では三次元ユーザインタフェースを用いて順序付き階層構造の閲覧を支援する手法 *Spiral Tree* を提案する。現在知られている階層構造の三次元視覚化手法 *Cone Tree* では属性の順序により整列された情報の先頭と末尾が不明瞭であり、整列されているという情報をユーザが利用しづらい。そこで *Spiral Tree* では、各階層のノードを円周上でなく螺旋上に配置することで先頭と末尾の位置を明示し、順序情報の利用を容易にする。

Spiral Tree の有効性を確認するために、*Cone Tree* との比較実験を行なった。さらに、閲覧を支援するシステムについて情報の閲覧作業に関する評価を行なうための「変更点探索タスク」を提案し、これを用いて閲覧作業の分析を試みた。その結果、このタスクは二つのサブタスクに分割して完了されることが分かった。また、定性的な結果は一部得られたが、定量的な評価のためにはタスクの難易度などを考慮した更に詳細な実験が必要であることが明らかになった。

A Method of 3D Visualization of Hierarchical Ordered Information

Takashi OKUMA, Haruo TAKEMURA, Yoshiaki KATAYAMA,
Hidehiko IWASA and Naokazu YOKOYA

Nara Institute of Science and Technology
8916-5, Takayama, Ikoma, Nara, 630-01

In this paper, we propose and evaluate a new method for 3D visualization of hierarchical ordered information. Cone Trees are known as a tool for 3D visualization of hierarchical structure. However, Cone Trees cannot visualize hierarchical ordered information effectively. It is hard to find the head and tail of ordered nodes, because they place each node on a circumference. To overcome the problem, we propose the Spiral Trees for 3D visualization of hierarchical ordered information. Spiral Trees place each node along a spiral and explicitly show the head and tail of ordered nodes clearly. Spiral Trees exploit interactive animation and 3D graphics in the same way as Cone Tree. In order to evaluate the effectiveness of the proposed method, an experiment with target node searching tasks was conducted. It was found that the Spiral Trees reduce the time required to search information. To analyze browsing with Spiral Trees, an experiment with error search tasks was conducted. It was observed that subjects divide an error search task into two subtasks. The result shows some qualitative characteristics of the Spiral Tree. However, for quantitative analysis, it was clarified that the difficulty of the tasks should be taken into consideration in the experiment.

1 はじめに

個人や企業が扱う情報は増加する一方であり、大規模情報の管理や利用を支援するツールへの要求が高まってきている。これを受けて、大規模情報の利用支援に関する研究が幾つか成されている [1, 2, 3, 4]。大規模な情報から特定の情報を引き出す際、引き出す情報が具体的に定まっていな段階では、情報全体を閲覧することで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。このことから、情報を利用するにあたって閲覧作業は重要な役割を果たすと考えられる。そこで、本稿では特に情報の閲覧を支援する手法について検討する。

情報の閲覧時には多くの情報が同時に見やすい形で提示されている方が望ましい。ところが二次元的に情報を配置して提示しても、限られた画面内ではすぐに限界に達する。そこで本稿では、奥行の表現を利用することで作業空間を拡大できる三次元ユーザインタフェースに着目する [5, 6, 7]。

閲覧によって引き出す情報を明確にしていく時、情報全体が内容によって分類されて階層的に管理されていれば大まかな分類から徐々に具体的な情報へと興味を絞っていくことでユーザは引き出す情報を具体化することができる。以上のことから大規模情報の閲覧を支援するツールとして、情報を階層的に管理し、三次元ユーザインタフェースで利用するものを考える。これを実現するためには、階層構造情報を三次元的に視覚化する必要がある。現在、階層構造の三次元視覚化手法として Cone Tree [8] が知られている。

本稿では、Cone Tree を用いて三次元の奥行き提示手法に関する実験を行なった過去の実験から [9]、情報の閲覧時における Cone Tree の問題点を指摘し、この問題点を解決する三次元視覚化手法として Spiral Tree を提案する。また、Spiral Tree における閲覧作業を評価する方法として「変更点探索タスク」を設定し、ドラッグや、アニメーションなどのインタフェースが順序付き階層構造の閲覧に与える影響を調べる実験、運動立体視や両眼立体視が閲覧に与える影響を調べる実験を行なったのでこれを紹介する。

以下、2 章では情報の閲覧作業について述べる。3 章では階層構造情報の視覚化手法である Cone Tree とその閲覧時における問題点について述べる。4 章では本稿における提案手法 Spiral Tree について説明する。5 章以降では Spiral Tree による閲覧作業の評価手法「変更点探索タスク」について述べ、これを用いて行なった閲覧作業を評価するために行なった予備実験について紹介する。

2 情報の閲覧

2.1 情報の閲覧作業

ユーザが情報を利用する場合、情報の検索だけを目的とすることは少なく、通常は他の大きな作業、例えば論文の執筆や報告書の作成などを目的とし、これを達成する手助けとして必要な情報を大規模な情報群から取り出して利用する。ユーザが情報を取り出そうとする場合、「出版社 A の書籍 B」というようにいつも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意識していな場合が多い。このような時に、情報に対して閲覧作業を行なうことで必要な情報にたどり着くためのきっかけを得ることができる。

本稿では閲覧作業を

- (1) 全体をさっと眺め (閲覧),
- (2) 興味のありそうな部分に注目し (注目),
- (3) 情報を引出し、内容を確認する (確認)。

という手順を構造全体に繰り返すことにより行なわれる作業であると定義する。

(1) の閲覧は三次元グラフィックスによる作業空間の拡大で支援することができ、(2) の注目に關しては階層構造による情報の管理で支援することができると考える。

2.2 情報の閲覧と階層構造

大規模な情報を関連するものにグループとしてまとめるという作業を繰り返すことで情報を階層構造として管理することができる。階層構造として管理されている情報を上の階層の自分の興味に近そうなものから閲覧していくことで全く関係の無い情報を確認してしまう確率を減らすことができる。このように階層構造で情報を管理することでユーザの閲覧作業時の負担を小さくできる。このように大規模な情報は、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、大規模な情報は階層的に管理されることが多い。

さらに、情報が持つ階層構造全体を視覚化することで、個々の情報を見るだけでは分からない情報、例えば情報同士の関連やユーザの作業に関連する情報の分布の状況などが分かるようになる。

3 階層構造の視覚化手法 Cone Tree とその問題点

Robertson らは、階層構造情報の三次元視覚化を行なうために Cone Tree を提案している [8]。Cone Tree は、一つの情報をノードと呼ばれる長方形のプレートで表し、情報の親子関係を半透明の円錐で表す。親ノードを円錐の頂点、子ノードを円錐の底の円周に均等に配置する。この円錐を階層的に配置することで階層構造情報を三次元視覚化する (図 1)。

Cone Tree では三次元グラフィックスとインタラクティブアニメーションを用いてユーザの認知的負荷を軽減し、大規模な階層構造情報の利用を可能にするといわれている。この利点は三次元ユーザインタフェースの基本的特質の側面をとらえるものとして重要だと考えられるが、実験などによる具体的な検証を Robertson らは行なっていない。

そこで、土本らはまずノード探索タスクによって、一般的な木構造の二次元視覚化と Cone Tree の比較実験を行なった [10]。その結果、ノード探索時間は二次元視覚化の

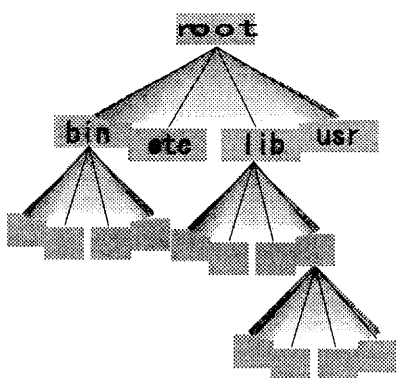


図 1: Cone Tree

方が短かった。しかし、探索対象の情報の数が多い場合においては、大部分の被験者が Cone Tree の方が操作しやすいと感じたとを土本らは報告しており、Cone Tree が大規模情報の操作感の向上に有効であるとしている。また、この実験に使用した Cone Tree は、アニメーションなどの設定が不十分であったと考察し、

- Cone Tree におけるアニメーションの方針と速度がノード探索タスクに及ぼす影響、
- 全ノード数、階層数、1 階層当りのノード数がノード探索タスクに及ぼす影響、

に関する評価実験を行い、この分析結果を反映した Cone Tree での再検討の必要性も指摘している [11]。土本らの実験において Cone Tree の方が二次元視覚化よりも検索時間がかかった原因は、アニメーションの設定ばかりでなく、視覚化した階層構造情報にも問題があったと考えられる。

一般に、情報には順序づけ可能な属性を持つものが多い。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」などがこれにあたる。ユーザはある情報を検索する時、この順序を手がかりに探索を行な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ため、各階層内の情報が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。このためユーザは順序付きの情報の探索に不便を感じる。土本らの実験で Cone Tree でのノード探索時間が二次元の視覚化よりも遅かったのは、実験に使用したデータが順序付きの階層構造であり、二次元視覚化では順序を探索の手がかりとして利用できたにも関わらず、Cone Tree ではうまく利用できなかったことも主要な要因の一つだと考えられる。

4 Spiral Tree

4.1 Spiral Tree の概要

Cone Tree において情報探索過程で順序情報が利用しづらいという問題点を解決するためには、順序に従って

整列された階層の先頭と末尾を明示する必要がある。これには、

- (1) 末尾と先頭の間に切れ目を入れる、
- (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。本稿では 3 の方法を採用する。3 を採用する理由は螺旋上に並べることによって各ノードの位置関係に円錐軸方向の「ずれ」が生じるが、情報の閲覧時には「ノードの位置のずれ」が「ノードの属性の順序」に直観的に対応しやすいと考えるためである。またそれ以外の理由としては、この手法が 1, 2 に比べて、

- 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、
- ノードの色を別の属性の視覚化に使用できる、

といった利点を持つからである。この方法で階層の先頭と末尾を明示するように Cone Tree を改良した視覚化手法を Spiral Tree と呼ぶ (図 2)。

Spiral Tree においても、三次元グラフィックスとインタラクティブアニメーションが用いられている。Spiral Tree において、ノードがマウスでクリックされると、クリックされたノードの属する階層が回転し、クリックされたノードが階層内で一番手前側になるように移動する。また、ノードをマウスでドラッグすることによりドラッグされたノードが属する階層を自由に回転させることができる。

4.2 Spiral Tree と Cone Tree の比較

提案手法の Spiral Tree の有効性を確認するために、Cone Tree との比較実験を行なった [12]。その詳細は以下の通りである。

4.2.1 比較実験

横置き型 Cone Tree (Cone Tree) のと、横置き型 Spiral Tree を実験用に作成した。表示した階層構造はルートノードの子ノード数 16、そのうち更に子を持つノードの数を 4 つとし、それぞれ 16 の子ノードを持つものとした (図 2)。詳細は以下の通りである。

- ノード：縦 0.6cm × 横 3.6cm
- 階層間距離：9.0cm
- 先頭ノードと末尾ノードのずれ (Spiral Tree のみ)：4.5cm
(注：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方向に平行移動して求められる。)
- 半径 一階層目：8.0cm，二階層目：3.1cm
- ノードの移動速度：1cm / 秒

以上の実験用 Spiral Tree および Cone Tree を使用して階層構造情報を被験者に提示した。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任意に抽出した単語であり、階層関係には特に意味を持たせていない。

実験は次の試行を繰り返すことで進めた。

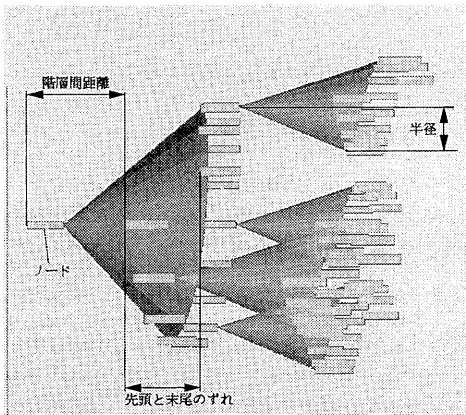


図 2: 評価実験に使用した Spiral Tree

- (1) 被験者に各階層のノード名をつなげたフルパス名で目標ノードをランダムに指定する。
- (2) 被験者は各 Tree を操作して目標ノードを選択 (マウスでクリック) する。

また、試行中も画面上部に目標ノード名を表示した。この際、指定された目標を選択するまでに要した作業時間とクリック数を測定し、比較することで表示法の違いが作業時間やクリック数に与える影響を調べた。また、実験後に被験者に対して 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を尋ねた。

試行数は Spiral Tree, Cone Tree とともに一人 100 試行、被験者は 6 名であった。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。

4.2.2 結果と考察

以上の実験から得られたデータについて 6 名の平均値に分散分析 ($\alpha = 0.01$) を行った結果、表 1 のような結果を得た。

この結果から以下のことがいえる。

- Spiral Tree を利用した場合は Cone Tree を利用した場合よりも、1 試行当りの探索作業時間が一秒近く短縮されており、探索作業時間の短縮に有効であることが分かる。
- 1 クリックあたりの探索作業時間が減少している事

表 1: Spiral Tree と Cone Tree の比較

	Spiral Tree	Cone Tree
平均時間 (秒)	5.68	6.57
分散	1.49	2.92
分散分析	有意差あり ($P < 0.01$)	
平均クリック数	1.34	1.38
分散	0.043	0.064
分散分析	有意差なし ($P = 0.24$)	
1 クリック当りの平均時間	4.24	4.76

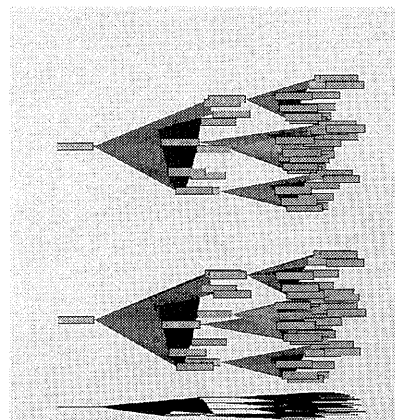


図 3: 変更点探索タスク中の画面写真

から、Spiral Tree による視覚化は順序付き階層構造でのノードの発見を容易にしていると考えられる。

また、被験者全員が Spiral Tree による視覚化の方が探索が行ないやすいと報告しており、操作感覚の向上にも有効であったことが分かる。

今回、クリック数の減少が確認できなかったのは、比較的単純な階層構造で実験を行なったため、Spiral Tree, Cone Tree のどちらもが最適なクリック数に近くなり、注目している階層の一覧性が増すという Spiral Tree の利点が顕著に現れなかったためだと考えられる。これについては今後、階層数の多い複雑な階層構造で同様の比較実験を行ない検討する必要がある。

5 Spiral Tree における閲覧作業の評価

5.1 変更点探索タスク

ノード探索タスクは、階層構造を操作する時の操作性の分析には有効であるが、閲覧作業の効率の定量的評価にはノード探索タスクは適さないと考えられる。これはノード探索タスクでは目標のノード位置を明確に与えるため、全体を閲覧するまでもなく目標のノードを選択することができるためである。

そこで全体を閲覧する必要がある実験タスクとして変更点探索タスクを設定する。このタスクは被験者に構造の違う二つの階層構造を提示して、その相違点を指示させるものである。二つの階層構造の相違点はある一つのノードの位置である。具体的には以下のようにして被験者に提示する二つの階層構造を作成する。

- (1) もととなる階層構造を用意する (Tree A)。
- (2) Tree A から葉 (子の階層を持たないノード) L と L の親以外の節 (子の階層を持つノード) M を一つずつランダムに選出する。
- (3) L を M の子の階層へと移動する。移動後の階層構造を Tree B とする。

試行中は、画面上部に Tree A、下部に Tree B を提示する (図 3)。Tree A, Tree B を閲覧しながら、それぞれにおけるノード L を被験者は指示する。この指示作業を 1 試行とする。Tree B は試行毎に Tree A から生成される。

このタスクは

- 閲覧の開始時においてどこに目標があるか分からない。
- 木の全体を閲覧せずには、タスクを達成することができない。

という二点が情報の閲覧作業と共通する。

被験者二名にこのタスクを行なわせ、その模様を観察した結果、両被験者ともこのタスクを次の二つのフェーズに分けて実行していることが分かった。

フェーズ1 Tree A/B どちらかの ノード L を閲覧によって見つけ出す。手順は以下のとおり。

- (1) 上下の階層構造の各々の階層を同じ角度にそろえて眺める。
- (2) 階層の形状から違いを感じた部分に注目する。
- (3) 注目した階層内のノード名を順番に見比べて行き、上下の階層構造で違うノードを発見する。

フェーズ2 L のノード名を頼りにもう片方の Tree の ノード L を探索する。各階層に対して以下の手順を繰り返す。

- (1) 形を眺め先頭と末尾の位置を確認する。
- (2) 先に見つけたノードの名前がありそうな場所に注目。
- (3) ノード名を確認

この観察から、フェーズ1とフェーズ2はほぼ独立した作業だと考えられ、定量的に評価する場合には分離して考える必要があると思われる。フェーズ1、フェーズ2の各々の作業が閲覧作業になっていると考えられ、このタスクが閲覧作業の評価に有効であると考えられる。

5.2 ドラッグとアニメーション

5.2.1 比較実験

Spiral Tree では階層情報の閲覧作業を支援するために、閲覧作業時の階層構造の操作方法としてドラッグによる階層の回転とクリックによる指定ノードの移動アニメーションを提供している。閲覧操作においてこれらの機能は重要であると考えられる。そこでこれが閲覧作業に与える影響を調べるために変更点探索タスクを用いた実験を行なった。

ドラッグによりドラッグされたノードが属する階層全体を自由に回転させることができる。また、アニメーションによりクリックされたノードが自動的にその階層の一番手前に移動するように属する階層全体が回転する。

ドラッグ、アニメーションのON/OFFの組合せから両方使用できない状態をのぞいた3通りで変更点探索タスクを行なわせてその様子を観察した。両方使用できない状態では手前のノードに隠れて指示できないノードが存在するので、実験を行なわなかった。被験者は二名、試行数は一人につき各状態40試行であった。

5.2.2 結果

各々のフェーズ終了にかかる時間を分析すると、分散が非常に大きかった。これは各試行のタスクの難易度の分散が大きいためだと考えられる。しかしながらドラッグのみで操作する場合に比べて、アニメーションを併用する場合の方ほとんどの場合で時間が減少しており、ア

ニメーションが有効であることを示していると考えられる(表3,4)。また、両方が使用可能になっているときの1試行当たりの平均利用回数は表2に示される通りである。

表2: ドラッグ/アニメーション平均利用回数

	被験者 N	被験者 I
アニメーション	2.8	1.8
ドラッグ	2.5	1.4

表3: 被験者 N のフェーズ1の終了平均時間(単位: 秒)

	自動回転あり	なし
ドラッグあり (分散)	24.39 (314.90)	38.51 (595.31)
ドラッグなし (分散)	28.67 (379.00)	— (—)

表4: 被験者 I のフェーズ1の終了平均時間(単位: 秒)

	自動回転あり	なし
ドラッグあり (分散)	25.47 (339.57)	31.58 (503.81)
ドラッグなし (分散)	21.48 (158.43)	— (—)

表5: 被験者 N のフェーズ2の終了平均時間(単位: 秒)

	自動回転あり	なし
ドラッグあり (分散)	10.83 (47.00)	17.59 (172.71)
ドラッグなし (分散)	10.12 (31.84)	— (—)

表6: 被験者 I のフェーズ2の終了平均時間(単位: 秒)

	自動回転あり	なし
ドラッグあり (分散)	12.80 (80.00)	10.49 (49.40)
ドラッグなし (分散)	9.58 (31.40)	— (—)

被験者の作業の模様の観察から、ドラッグやアニメーションの使用ができないことを告げているにも関わらず、現在使用できない操作方法を利用しようとするのがよく見られた。このことから両方の操作方法を提供することがユーザにとって自然であることを示すと考えられる。また、アニメーションは主に奥に隠れて見えにくいものを素早く手前側に移動させたい時に使用され、ドラッグは

- 上下の木の比較する階層を同じ向きにそろえるとき
 - 先頭から順にノード名を確認するとき
- 等に利用されることが分かった。

5.3 運動立体視と両眼立体視

5.3.1 比較実験

運動立体視とは頭部の位置を測定し視点の位置に合わせた透視投影を提示することで奥行きを提示する手法であり、両眼立体視とは液晶シャッター眼鏡などを用いて左右の目に視差を与えた透視投影を提示することで、奥行きを提示する手法である。これらの奥行き提示手法を使用して閲覧作業を行なうことから起こる影響を調べるため、運動立体視、両眼立体視のON/OFFの組合せによる

4 通りで変更点探索タスクを行なった。試行数は各々20であり、被験者は前述の実験に参加した被験者Nである。

5.3.2 結果

フェーズ1, フェーズ2ともに運動立体視の有無, 両眼立体視の有無について, 終了時間の平均値は運動立体視や両眼立体視が与えられている場合の方が時間がかかっている(表7,8)。この結果はCone Treeを用いたノード探索タスクによる実験結果[9]とは逆になっている。この要因としては

- 被験者がそれぞれの立体視に慣れていなかった,
 - 変更点探索タスクはノード探索タスクに比べてノードを閲覧する割合が多くなり, 焦点距離を移動しなければならぬ両眼立体視は疲労しやすかった。
 - Cone Treeでは奥になって隠れていたノードがSpiral Treeにおいては螺旋上の配置のために既に見えているので, Cone Treeに対する運動立体視の有効性の方がSpiral Treeに対する有効性より大きい。
- 等, 様々なものが考えられ, 今後検討していく必要がある。なお, タスク終了後に被験者Nに感想を求めたところ,
- 運動立体視があると画面全体が動くために上下を見比べにくい上に, カーソルでの指示が難しい。
 - 両眼立体視は疲れる。
 - 両眼立体視で立体的に表示されるとカーソルの動きの予測がつきにくい。
- と報告した。

今回, 分散値が大きく定量的な解析結果を得ることはできなかった。これはタスク毎の難易度に差がありすぎることなどが原因として考えられ, 今後更に詳細な実験が必要であると考えられる。

表 7: 被験者Nのフェーズ1の終了平均時間(単位: 秒)

	両眼視	単眼視
運動視あり (分散)	32.32 (583.33)	31.55 (356.67)
運動視なし (分散)	30.63 (541.65)	23.24 (239.11)

表 8: 被験者Nのフェーズ2の終了平均時間(単位: 秒)

	両眼視	単眼視
運動視あり (分散)	15.08 (91.52)	13.62 (60.39)
運動視なし (分散)	9.79 (17.58)	9.43 (30.17)

6 むすび

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について, 従来のCone Treeの改良型であるSpiral Treeを提案し, 順序を利用する検索に有効であることを比較実験により確認した。また, 階層構造の閲覧を評価する手法としての変更点探索タスクを提案し, 予備実験としてアニメーションやドラッグが閲覧に与える影響や, 運動立体視, 両眼立体視が閲覧に与える影響などの分析を行なった。この結果, 定性的な知見は得られたが, 定量的な結果を得るには至らなかった。

今後の課題としては, 被験者数を増やしての閲覧作業効率の評価, 階層数の多い構造を視覚化した場合の閲覧作業への影響の調査等が挙げられる。

謝辞

本研究の一部は文部省科学技術研究費補助金, 課題番号07780328により実施した。

参考文献

- [1] Ahlberg, C. and Shneiderman, B.: "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays," *Human Factors in Computing Systems(CHI '94)*, pp.313-317, (1994).
- [2] Rao, R. and Card, S.K.: "The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information," *Human Factors in Computing Systems(CHI '94)*, pp.318-322, (1994).
- [3] 飯沼, 三輪, 西山, 福永, 高堂, 上野, 小木, 柳川, 森田: "仮想空間を用いた情報蓄積・検索手法", *SICE HI News and Report Vol.10 No.2*, pp.205-210, (1995).
- [4] 水野, 青木, 辻: "データ可視化技法を用いた情報検索方式の提案", *SICE 第9回 HI シンポジウム集*, pp.79-82, (1993).
- [5] Rao, R., Pedersen, J.O., Hearst, M.A., Mackinlay, J.D., Card, S.K., Masinter, L., Halvorsen, P.-K. and Robertson, G.G.: "Rich Interaction in the Digital Library" *Commun. the ACM*, Vol.38 No.4, pp.29-39, (April 1995).
- [6] Robertson, G.G., Card, S.K. and Mackinlay, J.D.: "Information visualization using 3D interactive animation," *Commun. the ACM*, Vol.36 No.4, pp.57-71, (April 1993).
- [7] 三輪, 植田, 西山: "3次元グラフィックスを用いたネットワーク型情報の多面的管理と可視化", *SICE 第8回 HI シンポジウム集*, pp.613-616, (1992).
- [8] Robertson, G.G., Mackinlay, J.D. and Card, S.K.: "Cone Trees: Animated Visualizations of Hierarchical Information," *Human Factors in Computing Systems(CHI '91)*, pp.189-194, (1991).
- [9] 大隈, 竹村, 片山, 横矢: "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 信学春季全大, A-264, (1995).
- [10] 土本, 竹村, 片山, 萩原: "階層情報の3次元表示に関する実験的評価", 日本ソフトウェア科学会第11回大会論文集, pp.277-280, (1994).
- [11] 土本, 竹村, 片山, 萩原, 横矢: "階層情報の3次元視覚化に関する評価", 情処研報, HI59-8, (1995).
- [12] 大隈, 竹村, 片山, 岩佐, 横矢: "順序つき三次元視覚化手法の提案と評価", 1995 情処秋季全大, 7U-3, (1995).