

# VRML の論理構造に基づく 3 次元画像検索エンジンの設計と実装

山田 秀秋<sup>†</sup> 上原 邦昭<sup>††</sup> 田中 克己<sup>†††</sup>

VRML はインターネット上で 3 次元仮想空間や 3 次元仮想物体の形状や動作を表現するための言語である。ユーザはオリジナルな 3 次元空間を創造し公開することはもとより、様々なサイトで公開されている 3 次元仮想物体どうしをハイパーリンクでつないだり、自分の 3 次元仮想空間の中に取り込んでしまうことも可能となっている。しかし、WWW の特性により、各サイトや個人が、各々個別に管理、運用しているため、ユーザは、ユーザの嗜好に沿った 3 次元仮想空間や 3 次元仮想物体を探し出すことが非常に困難である。そのため、WWW 上の 3 次元画像情報を集約的に扱える仕組みが要求されている。そこで、本稿では、VRML 論理構造を用いた、VRML キーワード検索、幾何学情報による検索、part-of 関係に基づく検索および位置関係検索などが可能な検索エンジンとその索引構造に関して述べる。また、WWW 上で公開されている VRML コンテンツを自動的に収集し、その VRML の論理構造に基づいた検索システムに関して開発中の試作システムの実装についても述べる。

## Design and Implementation of a 3D Image Search Engine Based on VRML Logical Structure

HIDEAKI YAMADA,<sup>†</sup> KUNIAKI UEHARA<sup>††</sup> and KATSUMI TANAKA<sup>†††</sup>

The Virtual Reality Modeling Language (VRML) allows to describe 3D objects and combine them into scenes and worlds on the Web. Users can build their original 3D worlds. Users can also create hyperlinks from their scenes to other scenes or objects, and can incorporate models created by other creators into their scenes. Currently, it is difficult to search favorable 3D objects or scenes from the Web. This paper describes search methods of VRML data (searching by keyword, searching by primitives of contents, searching by part-of relationship, and searching by objects positions) and describes its index scheme. This paper also describes our prototype implementation of a VRML Search Engine based on VRML logical structure.

### 1. はじめに

現在、WWW 上で 3 次元仮想空間や 3 次元仮想物体の形状や動作を表現する言語として、Virtual Reality Modeling Language (VRML) が注目されている。VRML は、3 次元仮想空間や 3 次元仮想物体を表現する言語であり、ISO/IEC JTC1/SC24 で国際標準化提案<sup>6)</sup>がなされるなど、標準としての地位を築きつつある。ユーザはベンダーから提供されている VRML ブラウザをサーバからダウンロードすることにより、

様々なサイトで団体や個人が公開している VRML の 3 次元仮想空間内を自由にウォークスルーすることが可能である。一方、クリエイターは様々な CG 作成ツールを使用して、オリジナルな 3 次元仮想空間や 3 次元仮想物体を VRML コンテンツとして作成することが可能である。ただし、現在のオーサリング環境では、クリエイターが最初からすべてのコンテンツを作成する必要がある。クリエイターが他のクリエイターの作成した VRML 形式の 3 次元画像を再利用したい場合、以下の方法が考えられる。

- VRML 形式のサンプルが公開されているホームページにアクセスし、手探りでリンクをたどり目的の VRML データを探す。
- CD-ROM 媒体などで提供される VRML データ素材集を購入し、その中から利用できるモデルを

<sup>†</sup> 通信・放送機構神戸リサーチセンター

Kobe Research Center, Telecommunications Advancement Organization of Japan

<sup>††</sup> 神戸大学都市安全研究センター

Research Center for Urban Safety and Security, Kobe University

<sup>†††</sup> 神戸大学大学院自然科学研究科

Graduate School of Science and Technology, Kobe University

☆ VRML の仕様は、The VRML Architecture Group (VAG) から公開されている。http://www.vrml.org/Specifications/

探す。

- VRML のメーリングリストやネットニュース等にメッセージを投稿し、そのメッセージに対するリプライを待つ。

これらの方法では、クリエイタの必要とする情報に到達することが容易ではないといえよう。そのため、WWW 上で個別に管理運営されているコンテンツを集約的に扱える仕組みに対する要求が予測される。このような要求に対する一手段として、様々な検索エンジンを使うことが考えられる。現在、yahoo!<sup>8)</sup>、AltaVista<sup>12)</sup>、InfoSeek<sup>11)</sup>に代表される文字情報を対象とした検索エンジンは多数存在し、一般へのサービスも行われている。しかし、VRML 等の3次元物体を含むマルチメディアコンテンツに対して、その内容に基づいた検索を行うサービスはほとんど行われていない。SGI が公開している検索エンジン<sup>7)</sup>のような VRML を対象とした検索エンジンも存在するが、あくまでも VRML コンテンツのリンク元となっている HTML 文書の内容に基づいた検索であり、VRML コンテンツの内容そのものを検索するものではない。そこで、我々は VRML データの持つ論理構造をもとにした検索エンジンに関する研究を行っている<sup>1)~3)</sup>。本稿では、VRML データの論理構造に基づいた検索のあり方について、また、VRML コンテンツを自動的に集め、VRML データを検索するシステムの設計および試作システムの開発に関して述べる。ここで、VRML 論理構造を用いた検索手法として、VRML キーワード検索、幾何学情報による検索、part-of 関係に基づく検索および位置関係検索を提案する。また、それらの検索を行うための索引構造に関して述べる。

## 2. VRML データの論理構造と検索

### 2.1 WWW 検索エンジン

現在、WWW 上で各々が独自に管理を行っているホームページ上のコンテンツを集約的に取り扱うための仕組みの1つとして、様々な検索エンジンが存在している。その中で最も一般的に利用されているものは、HTML 文書を対象とした検索エンジンである。これらの検索エンジンは、大別すると下記の2通りのタイプが存在する。

- (1) 登録した文書から検索するサービス  
これらは、主に人間が介在し、登録した情報を整理し、解説をつけている。その中で代表的な検索エンジンとして yahoo!<sup>8)</sup>、JOY<sup>9)</sup>、galaxy<sup>10)</sup> などがある。
- (2) ロボット (プログラム) が自動的に集めた情報

を検索するサービス

一般にロボットと呼ばれるプログラムが自動的にハイパーリンクをたどり、芋蔓的に HTML 文書を取り込み、全文検索用の索引情報を生成する。HTML 中の `<META name="..." content="...">`、`<title>...</title>`、`<a href="...">...</a>` などのタグにランク付けを行い、そのランクに応じてスコアをつけることにより、検索結果の絞り込みを行っている。その中で代表的な検索エンジンとして AltaVista<sup>12)</sup>、Lycos<sup>13)</sup>、Infoseek<sup>11)</sup> などがある。

次に、HTML 文書以外の検索エンジンとしては、SGML 文書や2次元画像を対象とした検索エンジンが存在する。酒井ら<sup>5)</sup>は SGML の多様性を持つ論理構造から統一した論理構造に変換することによる索引生成手法の提案をしている。しかし、SGML では論理構造の決め方や文書そのものの記述の仕方のみが定義されているため、多様な SGML 文書の論理構造からの変換はルールが複雑であり高い処理コストを要する。坂内ら<sup>4)</sup>は画像を量子化した各色の頻度ヒストグラムを用いて検索を行う検索エンジンの開発を行っている。しかし、画像データはデータ内に論理構造を持たないため、統計的な手法での検索しか行えない。一方、VRML は3次元の物体、空間を表現する言語であり、その言語仕様として物体の3次元位置情報や物体間の包含関係などの論理構造を持っている。そのため、それらを意識した検索を行うことにより、より有用な検索が可能であると考えられる。たとえば、現実に即した名前付き部品や、部分・全体関係を持った部品の組合せ、またそれらの位置関係など、構造化情報としての様々な特徴を利用することができる。

### 2.2 VRML データの論理構造

VRML データは、たとえば図1に示すようにテキスト形式で書かれたデータであり、これによって3次元物体 (オブジェクト) の形状や振舞いなどが表現されている。それらのオブジェクトはノードと呼ばれ、ノードはフィールドとイベントから構成され、データはそれらの中に蓄えられる。ノードはシーングラフといわれる part-of 関係に集約される。図1の例をシーングラフで表すと図2となる。

VRML の特徴として、part-of 関係によるノードの集約とインスタンス化によるノードの再利用がある。前者は、仮想空間内に配置するオブジェクトを part-of 関係で集約し、全体としてのシーングラフを構成するという特徴である。たとえば、図1の例では、Desk ノードが Board, Column ノードを集約している。後

```

http://vrml.kobe-sc.tao.or.jp/sample/desk1.wrl
1: #VRML V2.0 utf8
2: PROTO TwoColorStool
3:   [field SFCOLOR boardColor .8 .4 .7
4:     field SFCOLOR columnColor .6 .6 .1 ]
5: {
6:   Transform {
7:     children
8:     DEF Desk Group {
9:       children [
10:        Transform {
11:          translation 0 0.6 0
12:          children
13:          DEF Board Shape {
14:            appearance Appearance {
15:              material Material {
16:                diffuseColor IS boardColor}
17:            }
18:            geometry Box { size 1.2 .2 1.2 }
19:          }
20:        }
21:        Transform {
22:          translation -.5 0 -.5
23:          children
24:          DEF Column Shape {
25:            appearance Appearance {
26:              material Material {
27:                diffuseColor IS columnColor}
28:            }
29:            geometry Cylinder { height 1 radius .1}
30:          }
31:        }
32:        Transform {
33:          translation .5 0 -.5
34:          children USE Column
35:        }
36:        Transform {
37:          translation -.5 0 .5
38:          children USE Column
39:        }
40:        Transform {
41:          translation .5 0 .5
42:          children USE Column
43:        }
44:      ]
45:    }
46:  }
47: }
48: TwoColorStool {
49:   columnColor 1 0 0 boardColor 0 1 0
50: }

```

図 1 VRML サンプル

Fig. 1 A sample VRML data.

者は、ノードにオブジェクト名をつけ、インスタンス(実体)化することにより、1つのVRMLファイル中で何度も使えるようにすることができるという特徴である。これには、DEF および USE が使用される。たとえば、図1の例では、DEF Column Shape{...}としてShapeノードをColumnと名前を付けてインスタンス化を行い、children USE Columnとして、一度定義したノードColumnを再利用している。

シーングラフは根を持つ順序付き木構造グラフ  $G(V, E)$  で表される。ここで、 $V$  は節点集合、 $E$  は枝集合である。各節点  $v$  は型を持ち、型ごとに異なった属性を有する。たとえば、型には Transform, Cylinder, Shape などがある。また、属性はその節点の性質を表すものと、節点間の接続関係や接続の種類を表すものに大別できる。VRML2.0における主な属性は以下のものからなる。

## (1) 節点の性質を表す属性

**translation** : Transform 型節点の属性で、平行移動を行う座標変換行列を表す

**scale** : Transform 型節点の属性で、スケーリングを行う座標変換行列を表す

**radius** : Cylinder 型節点の属性で、円柱の半径を表す

## (2) 節点間の接続関係や接続の種類を表す属性

**children** : Transform 型節点の属性で、その節点の子節点群を表す

**appearance** : Shape 型節点の属性で、見え方を表す Appearance 型節点との接続関係を表す

**material** : Appearance 型節点の属性で、材質を表す Material 型節点との接続関係を表す

本論文では、シーングラフの節点や枝に関する情報を次のように表している。節点  $v$  の全属性は  $fields(v)$ 、節点  $v$  の型は  $type(v)$ 、節点  $v$  の名前は  $name(v)$ 、節点  $v$  の属性  $a$  の値は  $v.a$ 、節点  $v$  の親節点に対する順番は  $order(v)$  で表している。

たとえば、図2において、節点  $d$  では  $fields(d) = \{ 'appearance', 'geometry' \}$ 、 $type(d) = 'Shape'$ 、節点  $g$  では  $g.height = 1$ 、節点  $b$  では  $b.translation = \{ 0, 0.6, 0 \}$ 、節点  $e'''$  では  $order(e''') = 5$  となる。

## 2.3 VRML 索引構造

本論文で示す検索エンジンは、多くのサイトに分散したVRMLデータを解析し、後の検索要求を迅速に処理するための索引を生成する。この索引レコードは、

(キーワード, 節点名, URL, 節点位置, 縮退シーングラフ)  
 $= (k, n, u, p, r)$

から構成される5項組レコードで定義される。ここで、キーワード  $k$  は節点名から抽出された単語、またはURL値中のファイル名から抽出された単語である。たとえば図1の場合、URL値 '<http://vrml.kobe-sc.tao.or.jp/sample/desk1.wrl>' から、'desk' がキーワードとして抽出され、また、図2の節点  $d$  では、節点名から抽出された単語 'Board' がキーワードとなる。節点名  $n$  は、キーワード  $k$  が抽出された節点  $v$  の名前である。たとえば、図2の節点  $d$  の例では、 $name(d)$  の値 'Board' が抽出される。ただし、URL中のファイル名からキーワード抽出を行った場合は、節点名を 'Root' とする。URL  $u$  は節点  $v$  を含むVRMLファイルのURL値である。たとえば、図1では、'<http://vrml.kobe-sc.tao.or.jp/sample/desk1.wrl>' が抽出される。節点位置  $p$  は根から節点  $v$  までの経路を示している。ただし、経路中の各節点は、節点型 [親

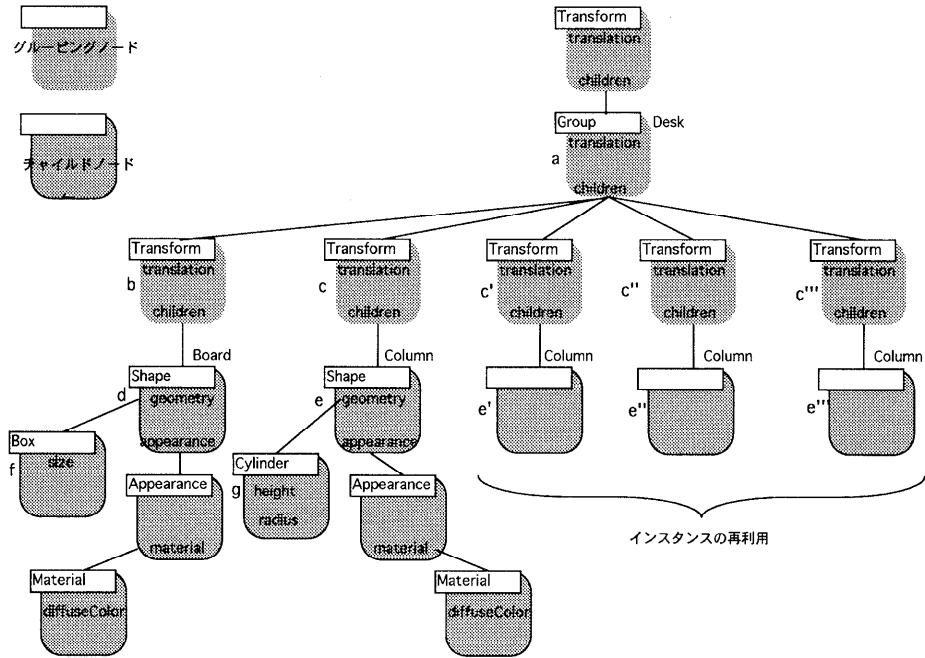


図2 シーングラフの例  
Fig. 2 A sample scenegraph.

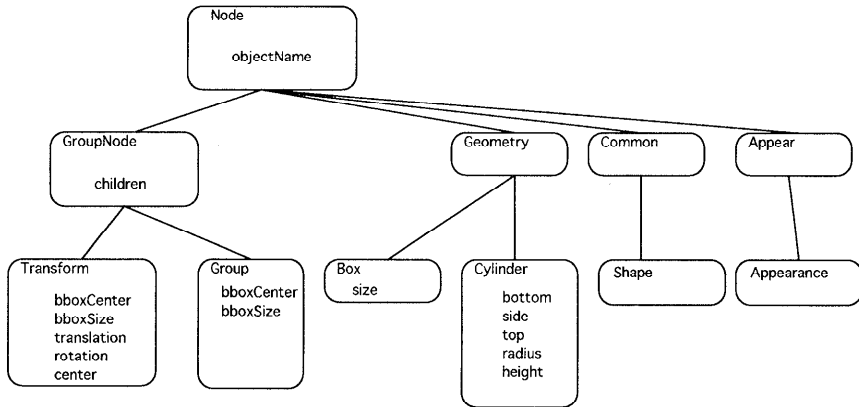


図3 縮退シーングラフのクラス定義  
Fig. 3 A hierarchical structure of a degrade scenegraph.

節点に対する順序] という形式で表現する. すなわち, 一般的には,

$$p = \text{type}(v_1)[\text{order}(v_1)] / \text{type}(v_2)[\text{order}(v_2)] / \dots / \text{type}(v_n)[\text{order}(v_n)]$$

となる. たとえば, 図2の根から節点cまでの経路は, '/Transform[1]/Group[1]/Transform[2]'と表される.

一般に VRML データには, ポリゴン物体の頂点情報, 色情報, 音情報, テクスチャー情報, イベント情報など, 本検索において不必要なデータが多数含まれている. そのため, シーングラフから検索に必要

な情報のみ残したものを縮退シーングラフとして定義している (図3). 縮退シーングラフ  $\mathcal{r}$  はキーワードが抽出された節点  $v$  を根とする部分木のすべての節点から **height**, **radius**, **transform**, **rotation**, **center**, **size**, **scale**, **scaleOrientation**, **bottom**, **side**, **top**, **bboxCenter**, **bboxSize** の属性値以外の属性値をすべて削除したグラフである. ここで, **bboxCenter**, **bboxSize** は, 物体の衝突検査やレンダリング処理における計算コスト削減のための物体を囲む境界箱 (boundary box) を設定している. したがって, 元の VRML データ中にあらかじめ値が入っ

ている場合はその値を利用するが、値がない場合は縮退シーン作成時に計算し付加している。

## 2.4 索引生成・検索アルゴリズム

### 2.4.1 索引生成アルゴリズム

与えられた VRML ファイルの URL  $u$  およびそのシーングラフから、下記の手順で索引を生成する。

#### Step1 URL からキーワードを抽出

索引レコードを新たに作成し、URL 中のファイル名をキーワードとして設定する。ここで、ファイル名や節点名はユーザが付けたものなので、普通名詞だけでなく数値や記号を付加したものが使われることが多い。たとえば、節点名 'Desk1' には数値 '1' が付加されている。このため、このステップでは [a-z], [A-Z] 以外の文字は削除し、純粋な単語として切り出す操作を行う。この処理をデータのクレンジングと呼ぶ。

**Step2** 節点  $v$  をシーングラフの根節点 'Root' とするシーングラフを根からたどるために、開始節点となる節点  $v$  を設定し、 $name(v) = 'Root'$  とし、節点位置  $p = '/'$  とする。

#### Step3 新規索引レコードの生成

VRML データ中に  $name(v)$  が定義されていれば、ここからキーワードを抽出し、クレンジング処理を経て索引レコードを新たに生成する。

#### Step4 索引用 URL の設定

対象とする VRML ファイルの URL から索引用 URL 文字列を生成し、索引レコードに設定する。ただし、'./sample/Room.wr1' のような相対的な URL は、統一的表記として絶対的な URL に変換したのち、索引レコードに設定する。

#### Step5 節点位置の設定

根から節点  $v$  までの経路を  $v$  の節点位置として索引レコードに設定する。

#### Step6 $v$ の縮退シーングラフの作成

$v$  を根とする部分木から縮退シーングラフを作成し、索引レコードに登録する。

#### Step7 次探索節点の設定

縦型探索で  $v$  の子節点を求め、これを新たに次探索節点とし、これを  $v$  として Step3 にもどる。もし、子節点がない場合には親節点の子節点を次探索節点とし、それを新たに  $v$  として Step3 にもどる。次探索節点がなければ終了する。

### 2.4.2 検索アルゴリズム

前項で生成した索引を用いて以下の 4 種の検索が可能となる。

#### (1) キーワード検索

節点名やファイル名を質問の条件とする検索である。たとえば、「机 (desk) を探せ」といった質問がこれに該当する。このような質問の処理は、検索用のキーワードを含む索引レコード群を検索し、これらに含まれる URL と節点位置の組集合を結果として返す。

#### (2) 幾何学的情報による検索

通常キーワードに加えて VRML の Box や Cone 等の幾何学形状もキーワードとして許した検索である。たとえば、「Box 形状のオブジェクトを含む Chair を求めよ」というような質問がこれに該当する。このような質問は、索引レコード中の縮退シーングラフを用いて以下の手順で処理される。

**Step1** 質問中で指定されたキーワードを  $k$ 、幾何学形状名を  $g$  とする。

**Step2** キーワード  $k$  を有する索引レコードを探す。

**Step3** Step2 でマッチした各索引レコードに対し、その縮退シーングラフを根から横型探索により、 $type(v) = g$  となる節点  $v$  が存在するか否かを探索し、もし存在すればこの索引レコードの URL と節点位置の組集合を解として返す。

#### (3) part-of 関係に基づく検索

部品間の部分・全体関係に基づく検索であり、たとえば「脚 (leg) が 4 本ある机 (desk) を求めよ」といった質問がこれに該当する。このような質問は、索引レコード中の縮退シーングラフ中に、節点間の part-of 関係情報が包含されており、これをもとに以下の手順で処理される。

**Step1** 検索キーワードを  $k_1$ ,  $k_2$  とし、 $k_1$  に  $k_2$  が含まれるものを求めるとする。

**Step2** キーワード  $k_1$  にマッチする索引レコードを探す。

**Step3** Step2 でマッチした索引レコードの縮退シーングラフを根から横型探索により、 $name(v) = k_2$  となる節点  $v$  が存在するか否かを探索し、もし存在すれば、この索引レコードの URL と節点位置の組集合を解として返す。

#### (4) 位置関係検索

複数の物体間の位置関係を条件として指定する検索である。たとえば、「コンピュータ (computer) が机 (desk) の上にあるシーンを求めよ」といった質問がこれに該当する。このよ

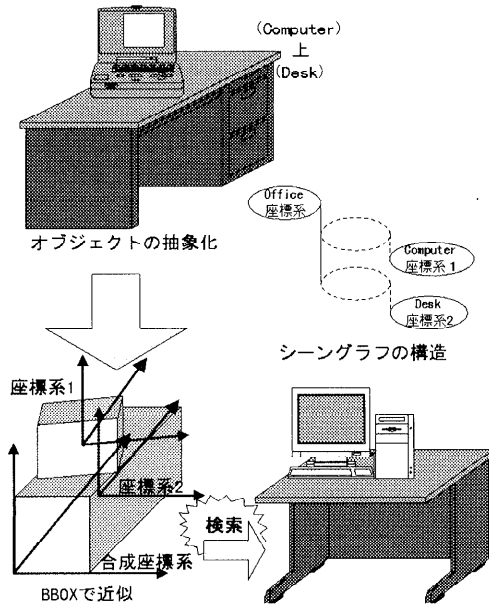


図 4 位置関係による検索

Fig. 4 A search method based on basic relations among contained objects positions.

うな質問の処理は、索引レコード中の縮退シーングラフ中に、節点間の part-of 関係とともに、境界箱の大きさや位置情報が包含されており、それらをもとに図 4 に示すように、上位節点からの各物体の境界箱間の位置関係を把握することができる。ここで、物体間の位置関係には基本的な位置関係概念<sup>16)</sup>を用い、節点  $v_1$  に対する  $v_2$  の関係  $r$  として次のものを用いる。

$$v_1 r v_2 = \{ \text{上, 下, 前, 後, 左, 右, 横, 内, 外, 奥} \}$$

この関係を使用し、上述のような質問は以下の方法で処理される。

**Step1** 検索キーを  $k_1, k_2$  とし、位置関係を  $r$  とする。

**Step2** キーワード  $k_1$  にマッチする索引レコードを探す。

**Step3** Step2 でマッチした索引レコードの縮退シーングラフの根  $v_1$  から横型探索により、 $\text{name}(v_2) = k_2$  となる節点  $v_2$  が存在するか否かを探索する。

**Step5** Step3 で求めた節点  $v_1, v_2$  の属性値  $v_1.\text{bboxCenter}, v_2.\text{bboxCenter}, v_1.\text{bboxSize}, v_2.\text{bboxSize}$  をもとに  $v_1, v_2$  の位置関係が  $v_1 r v_2$  となるか否かを判定し、もし存在すればこの索引レコードの

URL と節点位置の組集合を解として返す。

VRML データ中には、さらに、表面属性情報がある Appearance ノードとして含まれており、材質、テクスチャー等の情報も含まれているが、本システムでは現在対象としていない。本稿で設計、試作中のシステムでは、上記のような VRML キーワード検索、幾何学情報検索、part-of 関係に基づく検索および位置関係検索のみを対象としている。

### 3. VRML 検索エンジンの設計と実装

本システムは、図 5 に示すように、VRML データ探索ロボット部、データベース部、およびユーザインタフェース部を含む検索部に大きく分かれている。

#### 3.1 VRML データ探索ロボット部

我々が開発した探索ロボットは、探索起点の HTML ページのリンク情報をたどり、ハイパーリンクで示された VRML データの抽出を繰り返して探索を行う。本研究で試作した VRML 探索ロボットは、Java アプリケーションとして実装している。本探索ロボットは、VRML テキストファイルだけでなく、gzip により圧縮された VRML ファイルも自動的に解凍し、VRML データ内から索引情報を抽出できるように実装している。

探索ロボットは以下のように動作する。

#### (1) HTTP に基づくデータ転送

あらかじめ用意した URL のリストを探索起点として、それぞれの URL に対応したファイルを HTTP により転送する。

#### (2) HTML 文書の解析

転送したファイルから  $\langle A \text{ HREF} = \dots \rangle$  や  $\langle A \text{ EMBED SRC} = \dots \rangle$  等で示されたハイパーリンク先を抽出し、それぞれのハイパーリンク先の URL を絶対的なパスの URL として次の検索対象としてたどり、芋蔓式に WWW 上を探索する。ただし、探索が爆発的に増加することを防ぐため、起点となる URL からのリンクの深さが規定の閾値を超えると、探索を停止するように制限している。ハイパーリンク先が VRML ファイルである場合、そのファイルを VRML データの解析対象とするとともに、そ

\* たとえば、“ $v_1$  が  $v_2$  の上にあるものをさがせ” ( $v_1$  上  $v_2$ ) という質問では、 $v_1.\text{bboxCenter}, v_1.\text{bboxSize}, v_2.\text{bboxCenter}, v_2.\text{bboxSize}$  より求められる  $v_1, v_2$  の親節点の座標系における  $v_1, v_2$  の境界箱の XZ 平面への正射影による 2 つの 2 次元画像領域に重なる部分が存在し、 $v_1$  の XY 平面への正射影の全頂点の Y 座標が  $v_2$  の全頂点の Y 座標より大きい場合、答えが真となる。

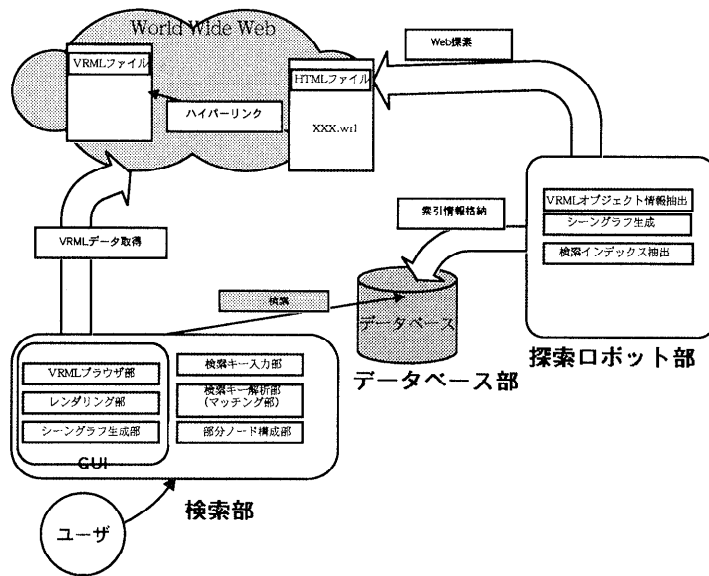


図 5 システム構成  
Fig. 5 Overview of a VRML search engine.

のハイパーリンクのテキスト情報やファイル名を VRML データへのキー情報として抽出する。

(3) VRML データの解析

転送した VRML ファイルを IICM の pw<sup>14)</sup> をもとに作成したパーザを用いて解析し、縮退シーングラフを生成する。ここで、pw は Java で実装された、GNU ライセンスに基づいてソースコードが公開されているフリーの VRML2.0 パーザである。VRML データの解析はノード単位で行い、各ノードのタイプに応じてフィールド情報を格納する。

(4) 取得したインデックスの格納

取得したインデックスをデータベースに格納する。

3.2 検索部

我々が設計試作中のシステムでは、以下の機能を有している。

(1) 検索結果のウォークスルー

VRML データを検索した検索結果も VRML データであり、検索結果の詳細を視覚的に確認するために、検索結果を VRML ブラウザでウォークスルーできる機能を有している。

(2) 検索結果のシーングラフに基づく検索結果の絞り込み

検索結果もオブジェクトであり、階層構造を持っているため、検索結果がユーザの期待する検索結果を子オブジェクトとして包含している可能性

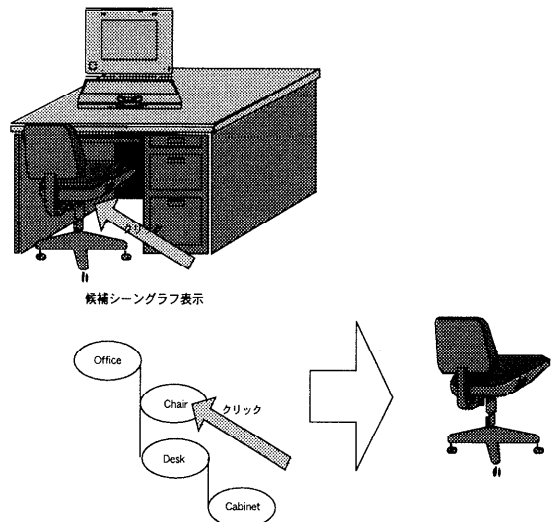


図 6 入力支援 GUI  
Fig. 6 A support GUI for input query.

がある。したがって、検索結果とそのオブジェクトのシーングラフを検索結果として提示し、シーングラフをもとに任意の子オブジェクトを抽出する機能を有している。

(3) グラフィカル・ユーザ・インタフェース (GUI)

3次元画像を検索するため、検索要求時においても検索式入力を GUI によりオブジェクト間の位置関係やオブジェクトの同定を支援する機能を有している。これは、図 6 に示すようにサンプルオブジェクトのクリックによるマウスイ

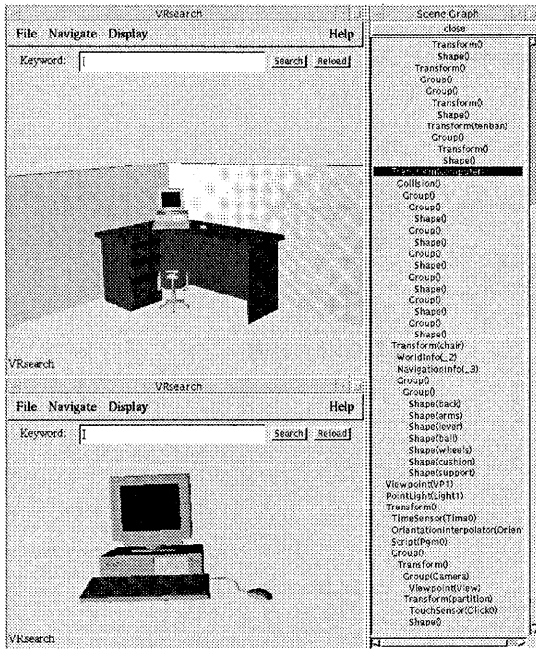


図7 ユーザインタフェース部  
Fig. 7 A part of user interface.

メントから候補シーングラフをサブ画面として作成し、シーングラフ上から選択することによりオブジェクトの同定を支援する機能である。

ユーザインタフェース部を図7に示す。試作したユーザインタフェースは、ユーザが入力した検索式よりVRMLデータを検索し、検索結果として返されるVRMLシーンを自由にウォークスルーするためのアプリケーションである。ユーザは検索結果として表れるVRMLシーンを図7左上部ブラウザウィンドウ上で自由にウォークスルーしたり、図7右部サブウィンドウに現れるシーングラフ中のノードをマウスでクリックすることにより、クリックされたノードに対応するVRMLオブジェクトを図7左下部ウィンドウに表示しながら、シーングラフをブラウザしたりすることができる。もちろん、図7左下部ウィンドウで表示されているサブシーンをウォークスルーすることもできる。図7の例では、図7左上部サブウィンドウに表示しているオフィス空間から、図7コンピュータを選択して図7左下のVRMLブラウザに表示している。本システムでは、VRMLブラウザウィンドウの一部、IICMで作成されたVRwave<sup>14)</sup>を使用している。これは、Java言語で作られたVRML2.0に対応したVRMLブラウザである。

### 3.3 実験結果および考察

本システムを用いて、The VRML Repository<sup>15)</sup>

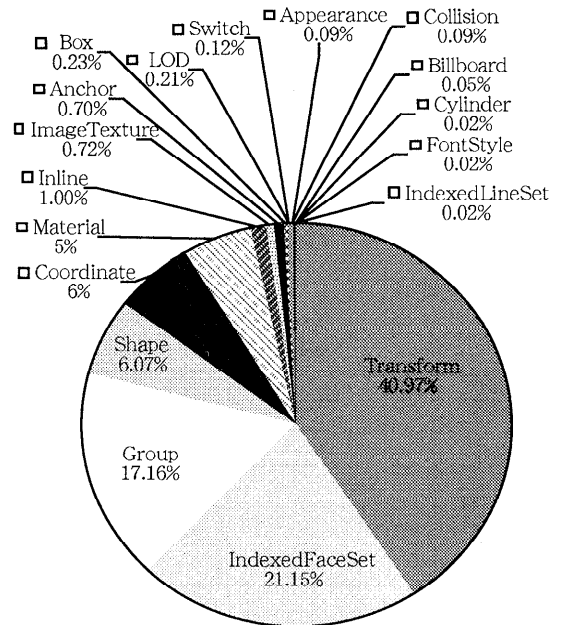


図8 ノードタイプによる分類  
Fig. 8 Classified results under nodetypes.

内のObject Librariesを探索起点として探索を行い、VRMLデータの収集およびインデックスの生成を行った。362件のVRMLファイルより、4313レコードのインデックスを抽出した。この実験により取得した1VRMLファイルあたりのファイルサイズは、非圧縮状態では1KBytesから5MBytesであるが、サイズの大きなファイルはgzip等で圧縮されているため、実際のダウンロードサイズは1KBytesから800KBytes程度である。1レコードあたりのインデックス生成にかかる時間は約3秒から30秒とかなりの幅があるが、ネットワークの遅延等を除くと約1秒程度である。また、検索スピードを決定するうえで最も重要な部分は縮退シーングラフの探索であるが、その縮退シーングラフのデータサイズはもとのVRMLデータのシーングラフの2%以下である。

本実験により取得したインデックスのノードタイプによる分類結果を図8に示す。この結果より、ユーザがインスタンスに名前付けを行っているノードは、6割がグルーピングを行うノードであり、2割が形状を表すノードであることから、仮想物体をVRMLファイル中で識別するために意識的にノード名が付けられていることが分かる。

キーワードの特徴に関する内訳を図9に示し、以下に各要素について述べる。

- 無加工で有効と思われるキー情報



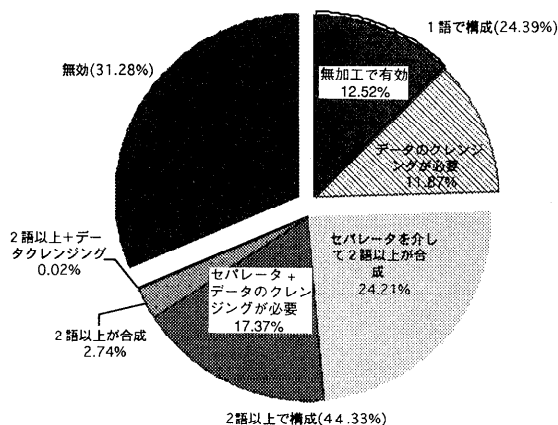


図9 キーワードの特徴による分類

Fig. 9 Classified results under feature of keywords.

castle, desk, lever, ball, chair, mat, frame, ...

これらは、そのまま検索キーとして有効である。

- セパレータで2語以上の構成となっているキー情報

Forward\_Hadron, RIGHT-THIGH, RightArm, ...

これらは、単純なフィルタで分離可能である。

- データのクレンジングが必要と思われるキー情報

TABLE1, leg0, ...

これらは、枝番号等が付与されているデータであり、データのクレンジング操作により対処可能である。

- 2語以上の構成となっているキー情報

MAINROOMFLOOR, sofasheet, ...

これらは、2語を分離する何らかの手段が必要である。

- 無効と思われるキー情報

0, v2%2, P000, G\_3500, PGM0, \_2\_4, obj\_1, shape\_1, ...

これらは、オーサリングツールやフィルタプログラム等で機械的にラベル付けされたままのデータと思われる。これらのキーは、ノード名による検索においては無効なインデックスである。

これらの結果より、7割の検索キー情報に対して、まったく2次処理を行わない状態でも有効なインデックスを生成できることが分かった。しかし、そのままの状態では、検索を行ううえで不十分なキー情報も多数存在するため、検索時、あるいは索引生成時に、2次処理を行うことにより、ユーザの検索要求に対して、よりたくさんの有益な検索解を求める工夫が必要と考

えられる。2次処理の手法としては、同義語辞書の導入、自然言語処理技術の導入、常識や知識の導入が考えられる。

#### 4. おわりに

本稿では、WWW上で提供されている3次元画像検索エンジンに関して、VRML論理構造に基づく検索手法、およびプロトタイプシステムの開発に関して述べた。本稿により、WWW上の3次元画像を集約的に取り扱うことが可能となった。本稿で述べたVRMLデータの検索は、基本的には索引構造のところで述べたキーワード名やオブジェクト名をもとに行うが、さらに、一般形として正規表現に基づく検索式表記を行えるように実装中である。これは、part-of関係をパス式で表記し、“複数の脚と引出しを持つ机”や“円柱でできた脚を持つ机”のようなpart-ofで表されるパス式の結合や属性を表現できるものである。このために、 $x, y, z$ のような変数を導入し、これをノードに対応させ、オブジェクト名を持つ変数 $x$ の定義を $name(x)$ として表記する。たとえば、Deskという名前を持つノードは $desk(x)$ のように表記する。また、“円柱でできた脚”のような属性値を持つ場合には、属性値をリストで表記し、 $x[a = \dots]$ と表記する。たとえば、ノードの属性 $geometry$ がCylinderであるLegというノードは $Leg(x)[geometry = Cylinder]$ として表記する。演算子としては、“ $\cdot, +, *, -$ ”を導入し、それぞれ“連結、和、閉包、無名変数”を表すものとする。パス式の結は、“ $\wedge, \vee, \neg$ ”を導入し、それぞれ“AND, OR, NOT”を表す。以上の定義を用いて“円柱の脚を2本以上持つ机”の検索式は、たとえば、

$$\langle Desk(x) \cdot * \cdot y \cdot Leg(w)[geometry = Cylinder] \cdot * \rangle$$

$$\wedge$$

$$\langle Desk(x) \cdot * \cdot y \cdot Leg(z)[geometry = Cylinder] \cdot * \rangle$$

$$\wedge$$

$$\langle z \neq w \rangle$$

のように表記する。これは、“名前がDeskであるノード $x$ の配下に、名前がLegであるノードが少なくとも2つあり、それぞれのLegの $geometry$ という属性がCylinderである”という意味である。

今後は、さらに多くのサンプルデータを収集し、これらのデータに関して分析を行い、その分析結果をもとに検索手法の改良を行っていく予定である。また、外部からデータベースを検索できるようにGUIの改良を行っていく予定である。

謝辞 本研究は、通信・放送機構神戸リサーチセンターにおける次世代デジタル映像通信に関する研究の

一環として行われた。この研究を行うにあたって有益なコメントをいただいた通信・放送機構神戸リサーチセンターの木邑信夫氏に感謝いたします。なお、著者の一部（田中，上原）は本研究において，一部，文部省重点領域研究（課題番号 08244103）および，日本学術振興会未来開拓学術研究推進事業における研究プロジェクト「マルチメディア・コンテンツの高次処理の研究」によっている。ここに記して謝意を表す。

### 参考文献

- 1) 山田秀秋, 木邑信夫, 田中克己: VRML 論理構造に基づく 3次元画像オブジェクトの検索と再利用, 情報処理学会研究会報告, 97-DBS-111, pp.119-124 (1997).
- 2) 山田秀秋, 木邑信夫, 田中克己: VRML 論理構造に基づく 3D オブジェクトの検索手法, 第 54 回情報処理学会全国大会論文集, Vol.3-73 (1997).
- 3) 山田秀秋, 木邑信夫, 上原邦昭, 田中克己: VRML 検索エンジンの設計と実装, 情報処理学会研究会報告, 97-DBS-113, pp.39-44 (1997).
- 4) 谷田部智之, 高羽洋樹, 佐藤 隆, 坂内正夫: 画像情報による情報媒介を目指した公開型画像サーチエンジン GIRLS, 情報処理学会, 情報シンポジウム論文集, Vol.96, No.10, pp.139-145 (1996).
- 5) 酒井乃里子, 高須淳宏, 安達 淳: SGML 文書データベースの論理構造変換における浮動要素の処理手法, 電子情報通信学会, 第 8 回データ工学ワークショップ (DEWS '97), pp.19-24 (1997).
- 6) <http://www.vrml.org/VRML97/DIS/ISO/IEC DIS 14772-1>
- 7) <http://vrml.sgi.com/>  
SGI の VRML 検索エンジンホームページ
- 8) <http://www.yahoo.com/>  
yahoo! のホームページ
- 9) <http://joyjoy.com/JOY.html/>  
JOY のホームページ
- 10) <http://www.einet.net/galaxy/>  
galaxy のホームページ
- 11) <http://www.infoseek.com/>  
InfoSeek のホームページ
- 12) Richard Seltzer/Eric J.Ray/Deborah S.Ray: The AltaVista Search Revolution, Osborne McGraw Hill, 1996,  
<URL:<http://www.altavista.digital.com/>>
- 13) <http://www.lycos.com/>  
Lycos のホームページ
- 14) <http://www.iicm.edu/vrwave/>  
IICM の VRML ブラウザのホームページ
- 15) <http://www.sdsc.edu/vrml/>  
The VRML Repository のホームページ
- 16) 岡田直之: 語の概念の表現と蓄積, 電子情報通

信学会 (1991).

(平成 9 年 9 月 1 日受付)

(平成 10 年 2 月 2 日採録)



山田 秀秋 (正会員)

1968 年生。1991 年九州工業大学情報工学部知能情報工学科卒業。1993 年同大学大学院情報工学研究科 (情報システム専攻) 修士課程修了。1994 年伊藤忠テクノサイエンス

(株) に入社。1996 年より通信・放送機構神戸リサーチセンター研究員として勤務。マルチメディア情報検索に関する研究に従事。



上原 邦昭 (正会員)

1954 年生。1978 年大阪大学基礎工学部情報工学科卒業。1983 年同大学大学院博士後期課程単位取得退学。大阪大学産業科学研究所助手, 講師, 神戸大学工学部情報知能工学科助教

を経て, 同大学都市安全研究センター教授。情報知能工学科を兼任。1989 年より 1990 年まで Oregon State University, Visiting Assistant Professor。1994 年より 1996 年まで神戸大学総合情報処理センター副センター長。工学博士。人工知能, 特に機械学習, マルチメディアデータベース, 自然言語によるヒューマンインタフェースの研究に従事。1990 年度人工知能学会研究奨励賞授賞。人工知能学会, 電子情報通信学会, 計量国語学会, 日本ソフトウェア科学会, システム制御情報学会各会員。



田中 克己 (正会員)

1974 年京都大学工学部情報工学科卒業。1976 年同大学大学院修士課程修了。1979 年神戸大学教養部助手。1986 年同大学工学部助教授。1994 年同大学教授 (情報知能工学

科)。1995 年同大学院自然科学研究科 (知能科学専攻) 専任教授, 現在に至る。主にデータベースの研究に従事。現在本会データベースシステム研究会主査。1996 年度より通信・放送機構「次世代デジタル映像通信の研究開発」の研究総括責任者, 文部省科研費重点領域研究「分散発展型データベースシステム技術の研究」の研究代表者。神戸マルチメディアインターネット協議会理事。人工知能学会, ACM 等各会員。