

## 視覚的対話機能を有する情報検索インタフェース

渡辺正裕 吉川正俊 石川佳治 植村俊亮

奈良先端科学技術大学院大学 情報科学研究科

本研究では、検索条件の入力や検索結果の出力を視覚的に行なう情報検索の新しいアプローチを提案する。このアプローチでは、キーワードや利用者に既知のオブジェクトを、直線または平面上に配置することにより、検索目標との相互位置関係によって視覚的に検索条件を指定する。検索結果として、検索目標の近傍のオブジェクトが視覚化して提供される。この過程を繰り返すことにより情報検索作業を進める。このような情報検索アプローチでは、キーワードのブール式などでは表現が困難な複雑な条件を視覚的かつ直観的に指定することができ、また検索条件の微妙な調整も行ない易い。

## Spatial Indication Mechanism for Exploratory Information Retrieval

Masahiro WATANABE, Masatoshi YOSHIKAWA, Yoshiharu ISHIKAWA and Shunsuke UEMURA

Graduate School of Information Science,  
NARA Institute of Science and Technology (NAIST)

We propose a new approach to information retrieval interface. In this approach, users specify retrieval conditions by spatially indicating keywords and/or objects in one or two dimensional space. The spatial relationship between target points and such keywords/objects is interpreted as search conditions. The system plots retrieved objects in the same space. With this approach, users can visually specify retrieval conditions which will be complicated when expressed in boolean expression of keywords.

## 1. はじめに

一般に情報検索を行なうときは、検索すべき目標がはっきりしている場合、どのような条件で検索すれば良いかが分かっている場合と、利用者本人にも自分が検索しようとしているものが何かよく分かっていないような場合がある。前者の場合は正確に問合せが発行でき、正確な検索結果が得られることが何よりも重要である。後者のような場合は、探索的に検索を進めるといった手探りの状態であり、利用者を取りまく情報空間や、利用者の検索動作に対する反応を一目で把握できることが重要である。

つまり、検索対象に関する利用者の知識の度合、検索意図などの状況に応じて有効な検索方法を選択できるのが理想であるといえる。

また、最近では大規模なハイパメディアやマルチメディアデータベースでの航行、拾い読み、検索が一般的なものになってきている。こういった目的で検索を行なうときにも利用者への応答が文字列で返ってくるシステムが多く存在する。これらの応答は末端利用者にとっては容易に理解できない場合があり、不親切なシステムであることがある。とくに手探りで探索する場合には、熟練した利用者にとっても文字列で表現された検索結果は直観的に把握しづらいものである。利用者は迅速に最小限の努力で、しかも情報空間で迷子になったりする恐れなしに、情報をフィルタにかけ、選択し、再構成することを望む。大規模データベースにおいて、情報を視覚化して提供することによって検索を容易なものにできると考えることができる。検索を行なう場合、利用者インタフェースは次のような点を考慮して設計されていることが望ましい。

- 利用者は自分の検索意図を、キーワードのブール式などのコマンド方式のみではなく、直観的にしかも簡単に指定できる。
- 利用者が検索時にキーワードのみならず、既知の具体的なオブジェクトも検索時の条件として利用できる。
- 検索結果を利用者に対して直観的に提示する機構を持つ。

本研究では、データベースに格納されているオブジェクトには $n$ 次元空間ベクトルが付随しているものとし、両者を同一視して議論する。これらのオブジェクトの中で利用者にとって既知であるものをいくつか選択して検索条件として利用する。 $n$ 次元のベ

クトルでは人間にとって直観的にとらえることが不可能であるので、これを1または2次元に射影することによってオブジェクト間の関係を視覚化し、その相互関係を利用者が直観的に把握できるようにする。これらを検索インタフェース空間上に配置し、その上で利用者が欲しいと考える検索目標を配置する。これらの相互位置関係から利用者の要求である検索目標の $n$ 次元ベクトルを推定し、その近傍のオブジェクトを検索結果とする。この検索結果を視覚化して提供しながら情報検索作業を進める情報検索の新しいアプローチを提案する。さらに電子図書館[ACM95]における文献検索を例にとって説明する。

2節では、関連する研究を紹介し、3節では本研究で用いるベクトルを定義する。視覚的な入出力インタフェースを考える上で4節と5節で入出力時における座標軸の有無に注目して分類する。

## 2. 関連研究

本研究の関連研究としては、2.1節、2.2節、2.3節のようなものがある。さらに続いて本研究のアプローチを実装する際に利用できる手法を2.4節、2.5節に挙げる。

これらの関連研究は、主に検索結果の表示を視覚化する研究であるが、本研究では、この射影した視覚化情報を結果の表示に用いるだけではなく、検索目標指定入力にも用いることを検討する。各関連研究とのそれ以外の相違点については、以下で個別に述べる。

### 2.1 Document Analysis for Visualization

この論文[Dub95]で紹介されているシステムでは、50~100の文献集合を視覚化するために、5~10個の単語を選択する。選択された単語を平面上に分散して配置し、それらの単語と各文献との類似度に基づいてその平面上に文献を視覚化して配置する。

このシステムはドキュメント空間の潜在的な構造を明らかにするという点で我々の提案する方法と同様である。しかし、検索条件を指定して直接文献を検索する機構を持たない点で、システムの目標に相違があるといえる。この研究では文献集合の内容に依存して視覚化を行なうが、我々の方法ではそれだけではなく、利用者の理解や世界観による個人差にも対応することを目指している。

## 2.2 Virtual Environments for Data Sharing and Visualization – Populated Information Terrains

この論文 [BM94] で紹介されている PITS では、単に共通のデータに対して作業するだけでなく、仮想データ空間に居住する複数の利用者が協調して作業をするのを支援しようとするのが目的である。PITS では共有されたデータの視覚化とそれに対する共同作業を支援する。その際、locking によって他者のアクセスを妨げるのではなく、お互いを情報作業空間で認識させることによって “social locking” を実現しようとしている。

我々の情報視覚化アプローチは、 $n$  次元を 2 次元に射影して視覚化するものであるが、この [BM94] ではディスプレイの 2 次元上に VR で 3 次元を表現し、さらに、大きさ、色、回転、振動などを利用することで数次元分多く表現しようとしている。PITS は協調作業を目指したシステムであり、検索を目的とした我々の提案する方法とは目標が異なっている。

## 2.3 SearchSpace: 2 次元空間へのキーワード配置を用いる全文検索システムのインタフェースと検索エンジン

全文検索システム SearchSpace [Tsu95][堤 95] では、従来型の AND/OR 条件式ではなく、複数のキーワードの 2 次元平面への配置で利用者の検索条件を表現する。平面上の位置はそのキーワードの優先度と曖昧さを表している。検索エンジンはキーワードの出現頻度に基づいて優先度を、文字列上の類似度で曖昧さを処理し、各文献への適合度を計算する。

2 次元への検索条件の配置という点で我々の提案する方法と同様であるが、SearchSpace では 2 次元平面に配置されるキーワード相互の相対位置関係を利用してない。さらに、我々は単語だけでなく利用者にとって既知の具体的な文献を配置することをも許している。これらの点でより柔軟であるといえる。

## 2.4 K-L 変換

K-L 変換 [森 90] とは、この方法をはじめて考案した Karhunen と Loeve にちなんで付けられた名前である。K-L 変換は、パターン認識の分野では古典的な手法であり、データが与えられると、これらを正射影した時に分散が最大になるように直交座標系を作りだす。つまり、データがもっとも散らばって見える方向に射影する。

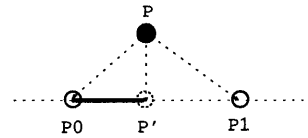


図 1  $n$  次元空間内におけるオブジェクト  $P$  の直線  $P_0 - P_1$  上への射影

## 2.5 FastMap

K-L 変換の通常の方法ではベクトル数  $N$  に対して  $O(N^2)$  の時間計算量が必要であり、大規模なデータベースに適していないなどの問題点が存在するが、これを近似的に線形時間で解決するために提案されたのが Faloutsos らによる FastMap アルゴリズム [FL95] である。

利用者は 2 次元平面上で検索目標を指定するが、これをもとに  $n$  次元の検索目標ベクトルを計算する。具体的には次の通りである。

利用者が検索目標オブジェクトを図 11 のように指定した場合、図 1 における  $(P_0P')$  は余弦定理を用いて、

$$(P_0P') = \frac{(PP_0)^2 + (P_0P_1)^2 - (PP_1)^2}{2(P_0P_1)}$$

となる。これが点  $P$  の直線  $P_0 - P_1$  方向を軸とした「座標」となる。さらに、図 2 において三平方の定理を用いて、

$$(P_hQ_h)^2 = (PQ)^2 - (X_p - X_q)^2$$

となる。これにより、 $n$  次元空間中の線分  $PQ$  を  $n-1$  次元超平面  $\mathcal{H}$  上に射影した線分  $P_hQ_h$  の長さが与えられる。

以上のステップを  $k$  回繰り返すことで  $P$  が乗っている  $k$  本の互いに直交する直線と、その方向を軸とした場合の点  $P$  の座標が求まる。

また、各  $n$  次元オブジェクト相互の距離が分かっていたら、オブジェクトを表すベクトルそのものが分からなくても良いということもこのアルゴリズムの特長となっている。

## 3. ベクトルの定義

本研究では、キーワードやオブジェクトに付随するベクトルの長さを正規化して用いることを前提と

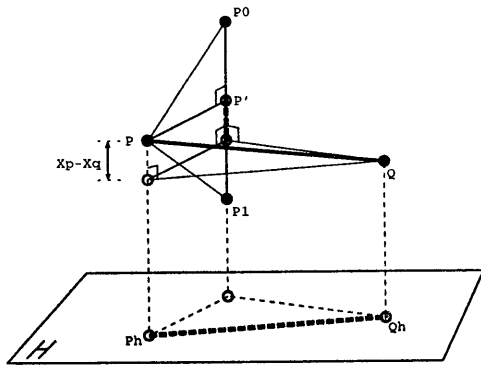


図2  $n$ 次元オブジェクトの  $n-1$ 次元超平面  $H$  上への射影 [FL95]

する。つまり、オブジェクトに付随する正規化ベクトルの終点はそのベクトル長さを半径とする超球面上に分布する。

### 3.1 ベクトルを作成する手法の例

$l$ 個の文献  $D_1 \sim D_l$ ,  $m$ 個のキーワード  $k_1 \sim k_m$ ,  $n$ 個の単語  $w_1 \sim w_n$ を仮定する。文献  $D_p$ に単語  $w_q$ が  $r$ 個含まれるとすると、文献ベクトル  $D_p$ の第  $q$ 成分を  $r$ とする。キーワード  $k_q$ を含む文献が  $D_{q_1}, D_{q_2}, \dots, D_{q_\alpha}$ であったとすると、キーワードベクトル  $k_q$ の第  $x$ 成分は  $D_{q_1}, D_{q_2}, \dots, D_{q_\alpha}$ の第  $x$ 成分の平均であるとする。さらに、入出力や検索に用いる文献ベクトルは、

$$d_p = \frac{D_p}{|D_p|}$$

のように正規化して用いる。

## 4. 座標軸を利用した入力方法

### 4.1 座標軸の定義

座標軸を用いて入力するには、1次元で入力する方法の他に、2次元で入力する方法、それ以上の次元で入力する方法が考えられる。例えば1次元で指定した後に2次元で指定するなど、複合的に繰り返し指定することができる。これらは組合せで議論できるため、ここでは1次元の場合について述べる。

座標軸は「ベクトル座標軸」と「スカラ座標軸」の2種類に分類して考える。

#### 4.1.1 ベクトル座標軸

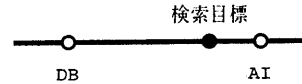


図3 ベクトル  $v_{AI}$ と  $v_{DB}$ を用いたベクトル座標軸

超球面上で2ベクトルの終点を最短距離で結ぶ経路を軸とする。この経路は曲線であるが、これを直線で表現したインタフェースを考える。この場合、 $v_{AI}$ というベクトルと  $v_{DB}$ というベクトルを選択すると、図3のようなインタフェースで検索目標ベクトルを指定することになる。これによって検索目標ベクトルを決定し、その近傍を検索結果とする。

#### 4.1.2 スカラ座標軸

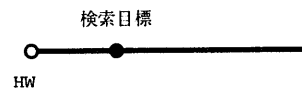


図4 ベクトル  $v_{HW}$ を始点としたスカラ座標軸

図4のようにある一つのベクトルの終点からの超球面上での距離を表す。例えば  $v_{HW}$ を始点とする線分インタフェース上で1点を指定すると、 $v_{HW}$ からこの点までの距離を半径とし、 $v_{HW}$ を中心とする超球面上の円周上付近を検索結果とする。

以下、既知のオブジェクトを4.2節ではシステムが自動的に配置し、4.3節では利用者が自由に配置する場合について議論する。

利用者に既知のオブジェクトを自由に配置することを許すと、様々な矛盾が生じ得る。そのような矛盾が生じないようにするためには既知のオブジェクトをシステムが自動的に配置しておけば良いと考えられる。

### 4.2 既知のオブジェクトをシステムが自動的に配置する方法

この方法では、ベクトル座標軸の場合、例えば  $v_{AI}$ というベクトルと  $v_{DB}$ というベクトルを選択すると、検索対象は、超球面上で  $v_{AI}$ ,  $v_{DB}$ を最短距離で結ぶ経路を検索範囲とする。さらに、 $v_{AI}$ と  $v_{DB}$ を結ぶ直線インタフェース上で検索目標ベクトルの位置を指定する。これにより、検索目標ベクトルを決定し、この近傍を検索結果とする。

利用者は図5や図7を用いてオブジェクトの指定を

行なうことができる。このとき利用しやすい方法として、

- システムがすべてのオブジェクトを図中に配置する
- 利用者がよく把握している、かつ、座標軸の特徴をよく表現する少数のオブジェクトに限定して配置する

という2通りが考えられる。

前者の場合、オブジェクトが十分に分散して配置されればそのまま指定入力を行なうことができる。分散の仕方が不十分であれば、領域を大雑把に指定して、徐々に限定してゆき、拡大した図で入力を行なう。

後者の場合、利用者がよく把握しているオブジェクトをシステムが知っていて、用いる座標軸の特徴をよく表現するオブジェクトを判断する必要がある。1次元なら座標軸の両端付近、2次元なら4隅付近のオブジェクトが座標軸の特徴をよく表現していると考えられる。システムはこれらとその間を適当な間隔で結ぶいくつかオブジェクトだけを表示し、利用者に入力してもらい。さらに、利用者は「判断に是非必要なオブジェクト」を適宜システムに要求し、図に追加させることができる。

- i) 1次元入力を用いて指定する手法（線分上のオブジェクトの指定）

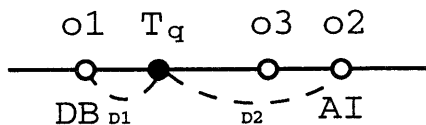


図5  $o_i$  と  $o_j$  を  $D_1:D_2$  に内分するようなオブジェクトの指定

この手法では、利用者がどういった尺度でオブジェクトを指定するか決定し、例えば尺度Aを基準にするとすると、それに応じてシステムは図5を利用者に提示する。図5において、 $o_1$ 、 $o_2$ 、 $o_3$  は利用者にとって既知のオブジェクトであり、座標軸の値に応じて線分図上に配置される。利用者はこの図5に対して検索オブジェクトの位置を指定する。

このような指定をスライドバーを用いて入力することもできる（図6）。

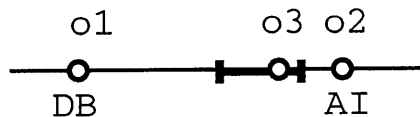


図6 スライドバー風入力方式の例

- ii) 2次元入力を用いる手法（平面上のオブジェクトの指定）

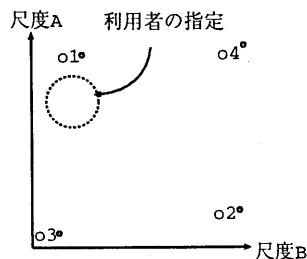


図7 2次元平面上でのオブジェクトの指定例

この手法では、利用者がどういった座標軸でオブジェクトを指定するか決定し、例えば尺度Aと尺度Bを基準にするとすると、それに応じてシステムは図7を利用者に提示する。さらに、4.1で説明したベクトル座標軸やスカラー座標軸を任意に組み合わせて用いることができる。図7は縦軸、横軸ともにスカラー座標軸を用いた例である。

この図において、 $o_1$ 、 $o_2$ 、 $o_3$ 、 $o_4$  は利用者にとって既知のオブジェクトであり、尺度A、尺度Bの値に応じて2次元平面上に配置される。利用者はこの図に対して検索オブジェクトの位置を指定する。

#### 4.3 既知のオブジェクトを利用者が自由に配置する方法

入力時には座標軸の方向にのみ意味を持たせ、長さの尺度は検索目標が入力されてから決定することとする。

利用者に既知のオブジェクトを自由に配置することを許すと、矛盾した指定が行なわれる可能性がある。この矛盾は利用者の理解を反映していると考えられるので、これを利用することを検討する。

$o_i, o_j$ を既知のオブジェクト,  $T_q$ を検索目標オブジェクトとすると, 矛盾は次のように分類できる.

- 検索する上ではあまり影響しないもの  
(例) 実際には  $o_i < o_j$  であるが, 利用者が  $T_q < o_j < o_i$ と指定した場合  
⇒ 利用者の理解を反映した矛盾であると考えられるので, 警告を発しておいてシステムが利用者がどう理解しているかを把握し, より効率的な検索に役立てる
- 検索する上で致命的なもの  
(例) 実際には  $o_i < o_j$  であるが, 利用者が  $o_j < T_q < o_i$ と指定した場合  
⇒ 入力し直してもらう必要があるが, これら入力されたオブジェクトについて利用者があまり詳しくないと判断できるので, この軸の方向に関しては重みづけを小さくするなどして検索に反映する

#### 4.3.1 1次元入力を利用する手法

「尺度 A について, 既知のオブジェクト  $o_i$  {より小, より大, 同程度}」といった指定を行なう. その際, 既知のオブジェクトも利用者がいくつか指定して, 検索目標の範囲を絞る手がかりとする. 4.2の i) の入力において, 検索目標以外に既知のオブジェクトをも指定させる.

#### 4.3.2 2次元入力を利用する手法

「尺度 A について既知のオブジェクト  $o_i$  {より小, より大, 同程度} で, 尺度 B について既知のオブジェクト  $o_i$  {より小, より大, 同程度}」といった指定 (図 8) を行なう. これを表現するために既知のオブジェクト  $o_i$  を入力座標平面の原点に配置し, A, B 各軸に対して第何象限に検索目標オブジェクトが配置されるかを入力させる. 次に  $o_j$  を原点に配置した座標平面で検索目標を指定させ, A, B 各軸に対して第何象限に検索目標オブジェクトが配置されるかを調べる. この指定を繰り返す (図 9) ことで A, B 方向の成分を指定する. 次に, C, D 平面で  $o_i$  に対する検索目標の位置を入力し, 続いて  $o_j \dots$  について C, D 方向の成分を指定する, といったように逐次各方向を指定してゆく.

以上の指定を  $m$  個の既知のオブジェクト  $o_1, o_2, \dots, o_m$  について行なわせる.

これらの指定を利用者に確認させるためには, 検索オブジェクトを原点に配置した図 10 を提示する.

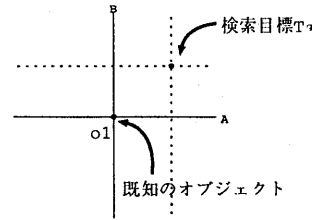


図 8 既知のオブジェクトに対する利用者の検索目標指定

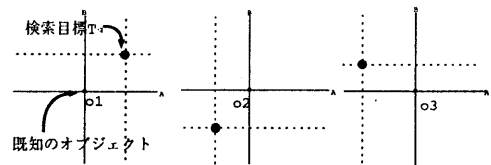


図 9 複数の既知のオブジェクトに対する利用者の検索目標指定

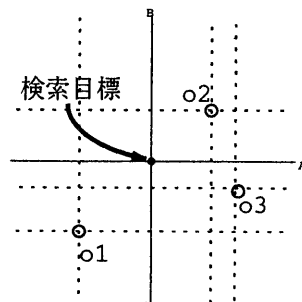


図 10 利用者による指定の確認

### 4.3.3 縮尺の決定

図 10において入力された既知オブジェクト ( $o_1, o_2, \dots$ ) 相互間の距離と、データベース中に格納されている距離を用いて、図 10の縮尺を決定する。

$n$ 次元の  $N$ 個のオブジェクトの像  $o_1 \sim o_N$ とそれらの間の距離情報  $Dist(o_i, o_j)$ をデータとして持つとする。そのうちの  $m$ 個が利用者によって図 10上に配置される。このオブジェクトを  $o_{k_1} \sim o_{k_m}$ とする。利用者がこの平面上に配置したオブジェクト相互の距離を  $dist(o_{k_i}, o_{k_j})$ とする。

$$(\text{縮尺}) = \frac{1}{mC_2} \sum_{i=1}^{m-1} \sum_{j=i+1}^m \frac{Dist(o_{k_i}, o_{k_j})}{dist(o_{k_i}, o_{k_j})}$$

この縮尺を用いて、検索目標  $T_q$ と各既知オブジェクトとの間の距離を決定することができる。この距離をできるだけ保持する（誤差が最小になる）ように検索目標  $T_q$ の位置を決め、その近傍のオブジェクトを検索結果とする。

## 5. 座標軸を利用しない入力手法

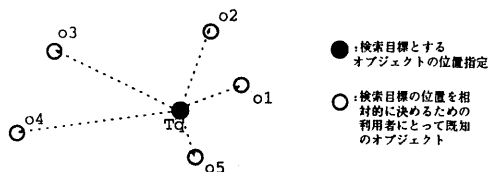


図 11  $n$ 次元オブジェクト  $P$ の位置を2次元平面上で指定した例

ここでは、利用者各個人がキーワードに対して持っている意味が異なる点を考慮し、 $n$ 次元意味空間を図 11のように2次元平面上に射影して利用する。目安となる（空間を定義する）文献やキーワードを利用者の好みの位置に配置できるものとする。

この手法の長所は、オブジェクトの配置の自由度が非常に高いことである。欠点は、座標軸がないために検索条件を微調整する際の手がかりがないことである。

## 5.1 検索目標ベクトルを計算する方法

### 5.1.1 各線分に射影する方法

任意の2オブジェクトの組で決定される直線を射影軸とすると、 $mC_2$  ( $< n$ と仮定する)本の軸が与えられる。この各軸に対する検索目標オブジェクトの座標をもとに $mC_2$ 次元の「超曲面」が構成できる。この「超曲面」からある距離以内のオブジェクトを検索対象とする。

### 5.1.2 データ中の距離でFastMapを施して利用者に返す手法

データ中の距離情報をもとに射影軸を選ぶ。これとデータ中の距離を用いて図中に入力された  $m$ 個の指定点をFastMapを利用することで2次元に射影し、利用者が検索目標ベクトルを指定する際の補助として、利用者に提供する。利用者はこれを参考にして改めて指定し直しても良いし、自分の理解に基づく配置のまま指定しても良い。

## 6. おわりに

本論文では検索条件の入力や検索結果の出力を視覚的に行なう情報検索の新しいアプローチを提案した。このアプローチによってキーワードや利用者既知のオブジェクトを、直線または平面上に配置して、検索目標との相互位置関係によって視覚的に検索条件を指定できるようにした。これによってキーワードのブール式などでは表現が困難な複雑な条件を視覚的かつ直観的に指定することができ、また検索条件の微妙な調整も可能となった。

表 1に、今回提案・比較したいいくつかの手法についてまとめた。

今後は、今回提案した方法の実験を行ない、その有効性を検証する予定である。

## 謝辞

本研究を進めるにあたって日頃から有意義な御指導・御討論をいただく植村研究室の皆様へ感謝いたします。

## 参考文献

- [ACM95] ACM: “Digital Libraries – Special Issue”, *Communications of the ACM*, Vol. 38, No. 4, April 1995.

表 1 座標軸の有無, 既知オブジェクト配置の自動・手動の方法の比較

座標軸あり		座標軸なし
方向に意味を持たせて指定するのが容易		方向に意味を持たせて指定するのが困難
既知オブジェクトをシステムが配置	既知オブジェクトも利用者が配置	既知オブジェクトも利用者が配置
矛盾が生じない	座標軸が決まっているので矛盾が生ずる可能性あり	既知オブジェクトの配置時の制約なし

- [BM94] Steve Benford and John Mariani: "Virtual Environments for Data Sharing and Visualizaion - Populated Information Terrains", in *Interfaces to Database Systems*, pp. 168-182, 1994.
- [Dub95] David Dubin: "Document Analysis for Visualizaion", in *ACM SIGIR '95*, pp. 199-204, 1995.
- [FL95] Christos Faloutsos and King-Ip(David) Lin: "FastMap: A Fast Alghorithm for Indexing, Data-Mining and Visualisation of Traditional and Multimedia Datasets", in *Proc. ACM SIGMOD International Conference on Management of Data*, pp. 163-174, 1995.
- [Tsu95] Fujio Tsutsumi: "SearchSpace: a Flexible Docment Retrieval System with an Interface Arranging Keywords in 2-D Space Expressing Fuzziness and Relevance", in *Proc. of Intl. Symp. on Digital Libraries 1995*, pp. 283-284, August 1995.
- [森 90] 森 俊二, 坂倉 梅子: 画像認識の基礎 [II], オーム社, 1990年.
- [堤 95] 堤 富士雄: "SearchSpace: 2次元空間へのキーワード配置を用いる全文検索システムのインタフェースと検索エンジン", 情報処理学会研究会報告, 95-FI-40, pp. 1-7, 1995年11月.