複数のメディアで構成された電子文書の検索手法

鈴木 優[†] 波多野 賢治[†]
吉川 正俊[†,††] 植村 俊亮[†]

近年，電子文書はテキストデータだけでなく，画像，映像など複数のメディアで構成されることが多くなった．したがって，これらの電子文書を検索するためには従来のように１つのメディアに特化した検索手法ではなく，複数のメディアに対応した検索手法が必要となる．本論文では，電子文書を構成している各メディアの評価値を統合することによって，複数のメディアで構成された電子文書を検索する手法の提案を行う．本手法では，電子文書から複数の特徴量をベクトルとして抽出し，利用者の問合せに含まれる各メディアのベクトルと比較を行うことによってそれぞれのメディアの評価値を求め，それらを統合することによって文書全体の評価値を求める．この検索手法は，複数のメディアによる問合せを行うことによって利用者の興味をより正確に検索システムに伝えることができるため，利用者の興味に応じた検索を実現することができる．また，図や文字を含むPDF 文書を用いて評価実験を行い，本手法の有効性を検証した．

A Simple and Integrated Retrieval Method for Multimedia Documents

Yu Suzuki,[†] Kenji Hatano,[†] Masatoshi Yoshikawa[†,††]
and Shunsuke Uemura[†]

Recently, electronic documents are composed of many kinds of medium such as images, videos, and so on. To retrieve these documents, we should consider not only text information but multimedia information. In this paper, we propose a method to integrate each score which is calculated by each retrieval method of the media. In our method, we extract documents' features such as term frequencies of text information, color histograms of image information, and layout information of these information, and generate feature vectors of each media from the media information. When users retrieve electronic documents, users search relevant electronic documents based on similarities which are calculated in each medium retrieval techniques. Therefore, users can retrieve electronic documents corresponding to the users' interest rather than text retrieval techniques. Furthermore, we evaluate our proposed method using PDF files, because PDF has a strict format concerned with layout information which is different from HTML's, and has not only text information but image information. We can verify the efficiency of our proposed method in these experiments.

1. はじめに

計算機の低価格化や高性能化，高速なネットワークの普及により，我々はテキストだけでなく画像や映像などの多くのメディアを扱うことができるようになった．したがって，これまでの情報検索のようにテキストデータだけを対象とするのではなく，それ以外のデータを利用した情報検索技術の重要性が高まっている．

従来のテキストデータだけを用いた情報検索では，問合せとしてキーワードを入力する検索方式が主であるが，電子文書ではテキストデータ以外にも画像データやそれらのレイアウトなどを利用した検索が考えられる．つまり，利用者の問合せとしてキーワードだけを入力するのではなく，より多くの手がかりを入力することによって問合せの表現能力を高めることで，利用者の興味に合致した電子文書を検索しやすくすることができると考えられる．たとえば，ある文字列や画像の位置を特徴量として抽出することができれば，利用者が以前見たことのある文書を検索する際に利用でき，従来の検索手法より利用者の検索要求をより忠実

† 奈良先端科学技術大学院大学情報科学研究科
Graduate School of Information Science, Nara Institute of Science and Technology
†† 国立情報学研究所ソフトウェア研究系
Software Research Division, National Institute of Informatics

に表現できるといえる．

そこで本論文では，文書から特徴量をできるだけ多く取り出し，利用者が検索する際の手がかりを増やすことによって文書検索の際の適合率の改善を図る．具体的には各メディアごとに特徴量を抽出し，それぞれのメディアの評価値を算出する．そして，それぞれの評価値を統合することで検索対象文書の評価値を求める．本研究は 1 つのメディアの評価値を統合する手法については他の研究と同様であるが，それらを組み合わせる方法，つまり評価関数についての議論が中心である．また，本提案の有効性を評価するための評価実験を PDF [2)] を用いて行い，従来の検索システムとの比較を行う．

本論文の構成は次のとおりである．2 章では基本的事項として PDF の概要と関連研究について述べる．3 章では本論文における電子文書からの特徴量抽出方法と特徴量のベクトル化，検索手法について論じる．4 章では本システムの評価を行い，5 章では本研究の結論および今後の課題について述べる．

2. 基本的事項と関連研究

本章では，本論文の実験で扱う電子文書のフォーマットである PDF について述べる．また，本手法との関連研究とその問題点についても述べる．

2.1 PDF

PDF が出現する以前，電子文書フォーマットとして PostScript [1)] が広く使われていたが，次のような短所があり，その保管や転送には不向きであった．

- 非常に大きなサイズのファイルとなることがある．
- 編集ができない．
- ファイル内の文字列を検索することはほぼ不可能である．

そこで，それらを解決するフォーマットとして PDF が Adobe 社によって考案された．PDF は画像や文字列だけでなく映像，音声も扱うこともできる．また，文書から画像だけを取り出すというように，文書の一部分だけを取り出すことが容易にできる．

こうした特徴から，現在大量の PDF 文書が使われているが，PDF 文書の検索は，テキスト情報だけを利用した検索だけが行われているのが現状である．

PDF の内部構造は大きく分けて 2 つの部分からなっている．

- ファイル構造（File Structure）
- 書類構造（Document Structure）

PDF では，文書をテキスト部，画像部などの部分（オブジェクト）に分け，それぞれの関係をファイル構造に記述する．

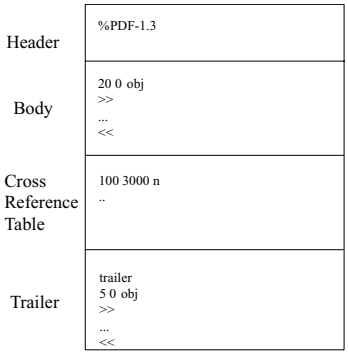

図 1 PDF の内部構造
Fig. 1 Structure of a PDF file.

2.1.1 ファイル構造

PDF は，**図 1** のような内部構造を持つ．

- **Header**
  `%PDF-<Version Number>` という記述．PDF のバージョン番号を表している．
- **Body**
  書類構造．2.1.2 項で詳しく説明する．
- **Cross-reference Table**
  オブジェクトのバイト位置情報を保持したもの．PDF ではオブジェクトを番号で示すため，番号でバイトの位置情報を参照するために用いる．
- **Trailer**
  `trailer` から `%%EOF` までの文字列．文書構造の root となるオブジェクト情報などを持っている．

2.1.2 書類構造

オブジェクトは**図 2** のような木構造になっている．

`Body` オブジェクトの例として，図 2 の `Catalog` オブジェクトの記述を**図 3** に示す．

“<<” から “>>” までがオブジェクトの内容であり，1 行目はヘッダ部分である．ヘッダ部分の先頭の文字 “1” はこのオブジェクトの番号であり，次の文字 “0” はこのオブジェクトが有効であることを示す．オブジェクトの内容は “/” で始まるコマンド列である．たとえば，“`/Type /Catalog`” は，このオブジェクトのタイプは `Catalog`（図 2 の root となるオブジェクト）であるという意味のコマンドである．“`Imageable content`” には，PostScript や JPEG 形式で書かれたデータが入っている．図 2 における “`Page`” の部分ではページ内のレイアウト情報が定義されている．多くの PDF ファイルでは `Catalog, Page tree, Page, Imageable content` だけで構成されているため，本研究で扱う PDF 文書はこれらのオブジェクトだけを扱う．

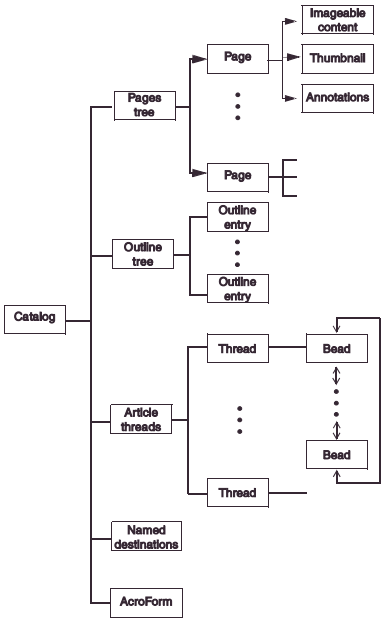

図 2 PDF のオブジェクト構造（文献 2) の 66 ページから引用）
Fig. 2 Structure of a PDF document (this figure is referred from Ref. 2) on page 66).

```
1 0 obj
<<
/Type /Catalog
/Pages 2 0 R
/Outlines 3 0 R
/PageMode /UseOutlines
>>
endobj
```

図 3 Catalog オブジェクトの記述
Fig. 3 Description of the Catalog Object.

2.2 関 連 研 究

従来の情報検索では，画像や映像など 1 つのメディアから多くの情報を抽出し，それを検索に利用することで検索精度を高める手法が主流であった．特に，画像の類似検索の研究では複数の種類の特徴量として色情報やテクスチャ情報（模様情報），形状情報などを組み合わせて検索する方法が数多くある[9]．これらの研究は，検索対象が 1 つのメディアに限定されているため，複数のメディアを検索対象としている本研究とは，扱うメディアの種類の点で異なっている．また，Web 上の画像を検索する際に Web 文書に含まれるテキスト情報を用いて検索を行う手法がある[4]．この手法では，まずテキストによる問合せでいくつかの画像の候補を検索し，それらの画像に類似した画像を画像検索手法で検索する方法である．この手法は画像の検索を容易に行うための手法であるが，利用者が一度画像の候補を絞る必要がある点，レイアウトによる問合せを行うことが非常に困難である点が本研究と異なる点である．

我々の研究のように既存の文書からレイアウト情報を取り出して，レイアウト情報を用いた検索手法がすでに提案されている[8]．この手法ではテキストとそのレイアウトを利用して検索する点は本研究との類似点であるが，テキストやレイアウトがベクトル表現されていないため，検索結果が順位付けされないという点が本研究との相違点である．同様に，既存の文書からテキストだけではなく画像などを取り出し，検索に応用しようとしている研究がある[5]．これらの研究と本研究との相違点として，これらの研究ではレイアウトやテキストを扱う特別な手法であり，音声や映像などを扱うにはまた別の手法が必要であるのに対して，本研究で示した手法はそのまま他のメディアに適用できる点があげられる．

また，文字列や画像のレイアウトを検索の手がかりとする研究も行われている．たとえば World Wide Web（WWW）で用いられている HTML 文書に含まれるテキストや画像を SQL を用いて検索する WebSSQL[7]では，SQL では定義されていなかった画像やレイアウトの問合せ方法を定義することで，問合せの表現方法の拡張を行っている．しかし，WebSSQL は SQL の改良であり，問合せ条件に適合した文書をすべて取り出すのに対して，本研究では適合する文書をランク付けして表示する点が異なる．さらに，複数のメディアの評価値を統合するための評価関数は，WebSSQL では最小値しか用いられていない．評価関数は検索精度に影響を与えるものであり，WebSSQL のように一意に定めるのは困難である．また，WebSSQL は HTML 文書を対象としているため，レイアウトが閲覧環境によって異なるといった問題もある．本研究ではこれらの問題点を考慮して，閲覧環境に依存しない電子文書形式である PDF を用いて実験を行い，6 つの評価関数を用いて実験を行っている．

3. 電子文書からの特徴量の抽出と検索への応用

本手法では，複数のメディアからなる電子文書から特徴量を抽出し，それらをベクトル表現する．本論文では，PDF 文書からテキスト情報やレイアウト情報，画像情報などできる限り多くの情報を抽出し，利用者

が検索するときの手がかりを増やすことによってより利用者に興味のある文書を検索できるシステムを構築することを目的としている．PDF 文書ではテキストだけでなく画像や映像，音声などを扱うことが可能であるため，それらを考慮した検索なども考えられるが，簡単のためテキスト，静止画像のみが出現する PDF 文書だけを考える．以下では複数のメディアで構成された文書からの特徴量を抽出する方法について説明する．

3.1 PDF 文書からの特徴量抽出

本研究で電子文書から抽出する情報は大きく分けて 3 つの情報（テキスト情報，画像情報，テキストや画像のレイアウト情報）である．本節では各情報の特徴ベクトルを定義する．

PDF 文書の集合を

$$DB = \{D_1, D_2, \cdots, D_l\} \qquad (l \geq 1)$$

と定義する．文書 D_i $(i = 1, 2, \cdots, l)$ を m_i 個のテキストオブジェクトや画像オブジェクトに分割し，

$$D_i = \{d_{i1}, d_{i2}, \cdots, d_{im_i}\} \qquad (m_i \geq 1)$$

と表現する．ここで d_{ij} $(j = 1, 2, \cdots, m_i)$ は D_i に含まれるオブジェクトであり，これらのオブジェクトから特徴量を抽出し特徴ベクトルを生成する．本研究では，テキストオブジェクトからは単語の出現頻度とレイアウトの特徴量，画像オブジェクトからは色ヒストグラムとレイアウトの特徴量を抽出する．オブジェクトの特徴量を次のように定義する．

$$\boldsymbol{f}(d_{ij}) = \begin{cases} [\boldsymbol{f}^{term}(d_{ij}), \boldsymbol{f}^{layout}(d_{ij})] \\ (d_{ij}\text{がテキストオブジェクトの場合}) \\ \\ [\boldsymbol{f}^{image}(d_{ij}), \boldsymbol{f}^{layout}(d_{ij})] \\ (d_{ij}\text{が画像オブジェクトの場合}) \end{cases}$$

ここで，$\boldsymbol{f}^{term}(d_{ij})$ はオブジェクト d_{ij} におけるテキストの特徴ベクトル，$\boldsymbol{f}^{image}(d_{ij})$ はオブジェクト d_{ij} における画像の特徴ベクトル，$\boldsymbol{f}^{layout}(d_{ij})$ は d_{ij} におけるレイアウトの特徴ベクトルをそれぞれ表している．以下では，これら特徴量を抽出し特徴ベクトルとして表現する方法について記述する．

3.1.1 テキストの特徴量抽出

テキストオブジェクトから単語の出現頻度や文の長さなど様々な特徴を特徴量として抽出することができるが，本論文では単語の出現頻度だけを用いた．まず，検索対象となる文書に含まれる単語を重複なく抜き出し，テキストベクトルの基底 W とする．

$$W = \{w_1, w_2, \cdots, w_n\}$$

特徴ベクトルの要素として，本研究では tf/idf 法[6] で重みを付けた単語の出現頻度を用いる．この方法を採用した理由として，重みに単語の出現頻度だけを利用した場合よりも良い検索精度が得られるからである[3]．DB に含まれる文書数を N，単語 w_k が含まれる文書数を df_k，オブジェクト d_{ij} における w_k の出現回数を tf_{ijk} とすると，テキストの特徴ベクトルは

$$\boldsymbol{f}^{term}(d_{ij}) = [f_{ij1}, f_{ij2}, \cdots, f_{ijn}]$$

と表現できる．ただし，f_{ijk} は次の式で計算される．

$$f_{ijk} = \frac{tf_{ijk} \cdot \log(\frac{N}{df_k})}{\sum_{k=1}^{n} tf_{ijk}}$$

3.1.2 画像の特徴量抽出

画像オブジェクトからは，色のヒストグラムやテクスチャ情報など様々な特徴を特徴量として抽出することができるが，本論文では色のヒストグラムだけを利用した．オブジェクト d_{ij} から各画素に対して画素値を計算した．d_{ij} 中の色番号が g $(0 \leq g \leq g_{\max}$：画像の色数) である画素の個数を c_{ij}^{g} と定義すると

$$\boldsymbol{f}^{image}(d_{ij}) = [\frac{c_{ij}^{1}}{c_{ij}}, \frac{c_{ij}^{2}}{c_{ij}}, \cdots, \frac{c_{ij}^{g_{\max}}}{c_{ij}}]$$

$$c_{ij} = \sum_{g=1}^{g_{\max}} c_{ij}^{g}$$

と表現される．つまり，画素値の割合を各特徴ベクトルの要素として考えている．

3.1.3 レイアウトの特徴量抽出

文字列オブジェクトや画像オブジェクトは，包囲矩形を用いて表現できる．レイアウトの特徴ベクトルは，オブジェクトを囲む矩形の座標を特徴量として抽出する．オブジェクト d_{ij} を囲む矩形の左上の座標を $(x_{ij}^{left}, y_{ij}^{left})$，右下の座標を $(x_{ij}^{right}, y_{ij}^{right})$ と定義すると，

$$\boldsymbol{f}^{layout}(d_{ij}) = [x_{ij}^{left}, y_{ij}^{left}, x_{ij}^{right}, y_{ij}^{right}]$$

と表現できる．

3.2 問合せの拡張

本手法では，次式で示された問合せを用いた検索を行う．$Query$ を問合せ，$Term$ を問合せとなる単語，$Image$ を画像に対する問合せ，$Layout$ を問合せとなる領域と定義すると

$$\begin{aligned} Query ::= & (\textbf{term } Term \textbf{ on } Layout) \\ | & (\textbf{image } Image \textbf{ on } Layout) \\ | & (\textbf{term } Term) \\ | & (\textbf{image } Image) \\ | & Query \ , Query \end{aligned}$$

と定義される．つまり，テキストや画像に関する指定，その位置情報という問合せを “,” で結んだものとして定義する．

問合せは大きくテキスト問合せと画像問合せに分けることができる．説明のため，テキスト問合せ，画像問合せそれぞれに番号 p，q を付与する．p 番目のテキスト問合せ特徴ベクトル $\boldsymbol{Q}_p^{term}$ $(p = 1, 2, \cdots, M_{term})$ は次のように定義される．

$$\boldsymbol{Q}_p^{term} = [t_p(w_1), t_p(w_2), \cdots t_p(w_n)]$$

ここで w_k は基底となる単語であり，キーワードが w_k と一致した場合は $t_p(w_k) = 1$ となり，一致しなければ $t_p(w_k) = 0$ となる．q 番目の画像の問合せ特徴ベクトル $\boldsymbol{Q}_q^{image}$ $(q = 1, 2, \cdots, M_{image})$ は次のように定義される．

$$\boldsymbol{Q}_q^{image} = [c_1, c_2, \cdots, c_{g_{\max}}]$$

ここで，c_g は画素の色 g の出現する割合である．

テキストのレイアウト問合せ領域となる矩形の左上端を (x_p^{left}, y_p^{left})，右下端を $(x_p^{right}, y_p^{right})$ とし，画像のレイアウト問合せ領域となる矩形の左上端を (x_q^{left}, y_q^{left})，右下端を $(x_q^{right}, y_q^{right})$ とすると，テキスト，画像のレイアウト問合せ特徴ベクトル $\boldsymbol{Q}_p^{layout}$，$\boldsymbol{Q}_q^{layout}$ は，

$$\boldsymbol{Q}_p^{layout} = [x_p^{left}, y_p^{left}, x_p^{right}, y_p^{right}]$$
$$\boldsymbol{Q}_q^{layout} = [x_q^{left}, y_q^{left}, x_q^{right}, y_q^{right}]$$

と表現される．もし利用者がレイアウトを指定しなかった場合は，$\boldsymbol{Q}_p^{layout}$，$\boldsymbol{Q}_q^{layout}$ は次のように表現される．

$$\boldsymbol{Q}_p^{layout} = [x_0^{left}, y_0^{left}, x_0^{right}, y_0^{right}]$$
$$\boldsymbol{Q}_q^{layout} = [x_0^{left}, y_0^{left}, x_0^{right}, y_0^{right}]$$

ここで，x_0^{left}，y_0^{left} は対象とする文書の左上の位置であり，x_0^{right}，y_0^{right} は対象とする文書の右下の位置である．

3.3 テキストオブジェクトの評価値の計算

本節では，文書全体の評価値を求めるための手順を説明する．まず，個々のオブジェクトに対する評価値を計算する．そして，それらの評価値を利用者の検索目的を反映し統合することによって，文書全体の評価値を求める．

3.3.1 テキストオブジェクトの評価値

テキスト部分の評価値は，3.2 節で求めた問合せベクトルと 3.1.1 項で求めた文書の特徴ベクトルとの類似度を求めることによって算出する．類似度に使用する関数としてコサイン相関値を使用する．つまり，あるベクトル$\boldsymbol{a}$，$\boldsymbol{b}$の類似度関数を sim とすると

$$sim(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a} \cdot \boldsymbol{b}}{|\boldsymbol{a}||\boldsymbol{b}|}$$

これをテキストの問合せとの類似度判定に利用する．テキストオブジェクト d_{ij} と問合せ $\boldsymbol{Q}_p^{term}$ との評価値 F_{ijp}^{term} は次のように表現される．

$$F_{ijp}^{term} = sim(\boldsymbol{Q}_p^{term}, \boldsymbol{f}^{term}(d_{ij}))$$

テキストオブジェクトのレイアウト部分は，問合せ領域とオブジェクトの占める領域との重なりを評価値として求める．問合せ領域 $\boldsymbol{Q}_p^{layout}$ の面積を $|\boldsymbol{Q}_p^{layout}|$，問合せ領域とオブジェクトの占める領域が重なっている面積を $|\boldsymbol{Q}_p^{layout} \cap \boldsymbol{f}^{layout}(d_{ij})|$ として，

$$F_{ijp}^{layout} = \frac{|\boldsymbol{Q}_p^{layout} \cap \boldsymbol{f}^{layout}(d_{ij})|}{|\boldsymbol{Q}_p^{layout}|}$$

と定義する．

これら 2 つの評価値を合成し，テキストオブジェクトの内容の出現頻度，レイアウトを考慮した評価値 F_{ijp}^{TL} を

$$F_{ijp}^{TL} = F_{ijp}^{term} \cdot F_{ijp}^{layout}$$

と定義する．

最後に，複数のテキストオブジェクトの評価値を合成し，文書全体のテキスト部分における評価値 X_i^{term} を求める．

$$X_i^{term} = \sum_{p=1}^{M_{term}} \left(\frac{\sum_{j=1}^{m_i} F_{ijp}^{TL}}{M_{term}} \right)$$

つまり，テキストオブジェクトの評価値の平均が高い文書ほど，相関が高いと考えられる．この理由としては，PDF 文書ではテキストオブジェクトがパラグラフ単位もしくは文単位で格納されているのではなく，物理的な位置によるまとまりであるためであることがあげられる．つまり，テキストオブジェクトは意味的なオブジェクト単位でまとまっているわけではないので，テキストオブジェクトを単位として文字列検索することに意味がない．これは，PDF が紙面の印刷，表示のみを考えてオブジェクトを配置しているにすぎず，論理的な構造を考慮していないからである．このような場合を考慮し，本研究ではテキストオブジェクトの評価値算出に相加平均を用いた．

3.3.2 画像オブジェクトの評価値の計算

画像部分の評価値は次のようにして求める．画像オブジェクト d_{ij} と問合せ $\boldsymbol{Q}_q^{image}$ との評価値 F_{ijq}^{image} は次のように表現される．

$$F_{ijq}^{image} = sim(\boldsymbol{Q}_q^{image}, \boldsymbol{f}^{image}(d_{ij}))$$

画像オブジェクトのレイアウト部分は，テキストオブジェクトにおける処理と同様，問合せ領域とオブジェクトの占める領域との重なりを評価値として求める．問合せ領域 $\boldsymbol{Q}_q^{layout}$ の面積を $|\boldsymbol{Q}_q^{layout}|$，問合せ領域とオブジェクトの占める領域が重なっている面積を $|\boldsymbol{Q}_q^{layout} \cap \boldsymbol{f}^{layout}(d_{ij})|$ として，

$$F_{ijq}^{layout} = \frac{|\boldsymbol{Q}_q^{layout} \cap \boldsymbol{f}^{layout}(d_{ij})|}{|\boldsymbol{Q}_q^{layout}|}$$

と定義する．

これら2つの評価値を合成し，画像オブジェクトの色ヒストグラム，レイアウトを考慮した評価値 F_{ijq}^{IL} を

$$F_{ijq}^{IL} = F_{ijq}^{image} \cdot F_{ijq}^{layout}$$

と定義する．

最後に，複数の画像オブジェクトの評価値を合成して，文書全体の画像における評価値 X_i^{image} を求める．

$$X_i^{image} = \sum_{q=1}^{M_{image}} \left(1 - \prod_{j=1}^{m_i} (1 - F_{ijq}^{IL}) \right)$$

なぜなら，画像は文書中に複数存在する場合があるが，1つの類似度の高い画像が含まれた文書の方が，多数のそれほど類似していない画像を含む文書よりも高く評価されなければならないからである．

3.4 文書全体の評価値

我々の目的は文書 D_i の評価値 X_i を求めることである．X_i は 3.3 節で求めた X_i^{term} と X_i^{image} という2つの異なるメディアの評価値を統合しなければならないが，本研究では統合するための評価関数を6つ定義し，実際に実験を行うことによって精度の高い評価関数を求めた．本研究で利用する6つの評価関数を以下に示す．

- 相加平均

$$X_i = \frac{X_i^{term} + X_i^{image}}{2} \quad (1)$$

この評価値を用いると，ただ1つのメディアの評価値が高くても，X_i の値はほとんど変化がない．

- 相乗平均

$$X_i = \sqrt{X_i^{term} \cdot X_i^{image}} \quad (2)$$

相加平均の場合と同様に，ただ1つのメディアの評価値が高い場合でも，X_i の値はほとんど変化がない．

- 調和平均

$$X_i = \frac{2}{\frac{1}{X_i^{term}} + \frac{1}{X_i^{image}}} \quad (3)$$

相加平均，相乗平均の場合と同様，ただ1つのメディアの評価値が高くても，X_i の値はほとんど変化がない．実際には X_i^{term} や X_i^{image} の値のどちらかが0である場合には0以上の非常に小さな値に置き換えた．

- 最大値

$$X_i = \max\{X_i^{term}, X_i^{image}\} \quad (4)$$

この評価値を用いると，X_i^{term} や X_i^{image} のある1つの評価値が高い場合でも高い評価値を得ることができる．よって，ある1つのメディアの評価値が高く他の評価値が低い場合でも，X_i の値は高くなる．

- 最小値

$$X_i = \min\{X_i^{term}, X_i^{image}\} \quad (5)$$

この評価値を用いると，X_i^{term} や X_i^{image} のある1つの評価値が低い場合に文書の評価値が低くなる．本研究の類似研究である WebSSQL [7]はこの関数を用いて評価値を算出している．

- PRO 関数

$$X_i = 1 - ((1 - X_i^{term}) \cdot (1 - X_i^{image})) \quad (6)$$

この評価値では，X_i^{term}，X_i^{image} のどちらか一方もしくは両方の評価値が高い場合に X_i の値が高くなる．

例として，3つの文書 D_1，D_2，D_3 を考える．各文書の文字列オブジェクト，画像オブジェクトの評価値はそれぞれ $X_1^{term} = 0.9$，$X_1^{image} = 0.5$，$X_2^{term} = 0.5$，$X_2^{image} = 0.5$，$X_3^{term} = 0.1$，$X_3^{image} = 0.5$ であったと仮定する．利用者はこれら 3 つの文書を $X_1 > X_2 > X_3$ と評価するのが自然であると考えられる．ここでシステムが 6 つの異なる評価値算出法（相加平均，相乗平均，調和平均，最大値，最小値，PRO 関数）を用いて評価値を算出することを考える．それぞれの評価関数で D_1，D_2，D_3 の評価値 X_1，X_2，X_3 を算出した結果を**表 1** に示す．

まず，最大値では X_2，X_3 は同じ評価値であることが分かる．だが，文書 D_1，D_2，D_3 は文字列，画像の評価値が大きく異なるので，文書の評価値へその違いが反映されておらず，良い評価値とはいえない．ま

表 1 6つの異なる評価関数を用いた評価値の違い
Table 1 Differences between six evaluation values.

	X_1	X_2	X_3
相加平均	0.7	0.5	0.3
相乗平均	0.671	0.35	0.22
調和平均	0.643	0.5	0.17
最大値	0.9	0.5	0.5
最小値	0.5	0.5	0.1
PRO 関数	0.95	0.75	0.55

た，最小値では X_1，X_2 が同じ評価値となることから，やはりテキスト部分の評価値の違いが文書の評価値に反映されていない．これらのような問題を解決するために，PRO 関数や相加平均，相乗平均などを評価値として用いることを考えることができる．確かにこれらの方法を用いると，利用者が適合していると仮定した順にランキングを行うことができるが，この例だけではどの手法が良いのか一般的に決めることができない．そこで 4 章では，これら 6 つの評価関数を用いた実験を行い，良いと思われる評価関数を求める．

4. 評価実験

本手法は既存の電子文書検索手法における問合せよりも多くの要素を問合せとして利用するため，テキストだけを問合せとする既存の電子文書検索システムと比較することができない．そこで，以下の 2 つの実験を行うことによって本手法の有用性を証明し，また評価関数を決定することにした．

(1) 問合せとしてテキストだけを指定した場合の本システムと既存のシステムとの検索精度の比較
(2) 本システム上でテキストだけを問合せとして用いた場合とすべての特徴量を指定した場合の検索精度の比較

これらの評価を行うことによって，問合せの自由度の異なる 2 つの検索システムを同じ条件で比較することができると考えられる．

以上のような実験を行うために，本手法を実装した．全体図を**図 4** に示す．

4.1 実験 1

4.1.1 実験方法

本研究で提案した手法が既存のシステムと比較して有効であることを確かめるために，本研究で提案した手法を実装したシステムと，既存のシステムとの比較実験を行った．実験で扱うデータとして，“2000 Digital Symposium Collection” に含まれる複数の会議録のうち，PODS，KDD，ICDE の 3 つの会議のものを使用した．ACM Digital Library との比較で用いた文書は PODS，KDD に含まれるものであり，100 文書である．IEEE Digital Library との比較で用いた文書は ICDE に含まれる 96 文書である．これらの PDF 文書群は論文の集合であるため，文字列だけではなく画像が混在しているため本手法が適用できる．また，実際に論文を検索をする場合に画像やその位置を問合せとする場合が十分考えることができるため，これらの文書群を扱うことによって実用性を証明することができると考えられる．以下に本実験の手順を示す．

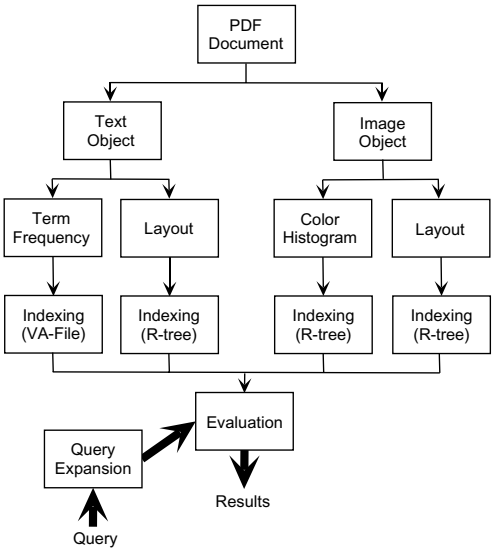

図 4 システムの全体図
Fig. 4 An overview of our proposed system.

表 2 それぞれの問合せに対する答えの数
Table 2 The number of relevant documents.

	Query (a)	Query (b)	Query (c)
ACM DL	5	5	5
IEEE DL	4	3	5

(1) 利用者の問合せを考える．本実験では
  (a) マルチメディア文書データベースに関する論文
  (b) XML の検索に関する論文
  (c) 多次元空間における索引に関する論文
  を問合せとして利用する．
(2) 問合せに対して，あらかじめ人手で正解集合を求めておく．
(3) 本研究で構築したシステムを用いて，2 つのシステムで答えを求める．
  (a) 本研究で提案した手法を実装したシステム
  (b) ACM Digital Library
  (c) IEEE Digital Library
(4) あらかじめ求めておいた解答集合と比較することによって，再現率–適合率グラフを求める．

本実験における問合せは，すべてのシステムである程度の数の結果が得られるものを選んだ．

今回のような問合せの検索結果は，情報が適合するかどうかという評価が主観的なものとなってしまい，評価者によって異なるものである．また，これら複数のメディアからなる文書が適合しているかどうかのテストコレクションは存在しないため，実験の際に作成

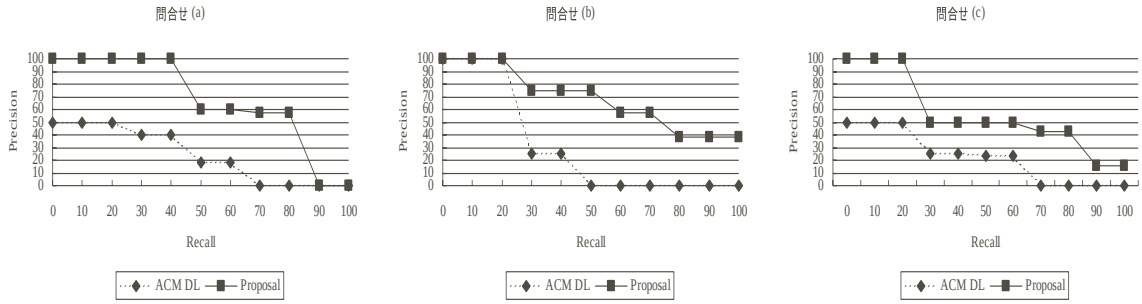

図 5 ACM Digital Library と本手法との再現率–適合率グラフによる比較
Fig. 5 Recall–precision graph of ACM Digital Library and our proposal system.

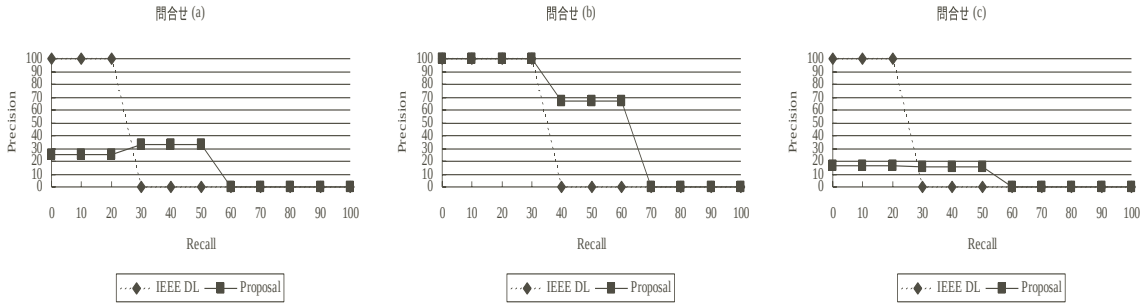

図 6 IEEE Digital Library と本手法の再現率–適合率による比較
Fig. 6 Recall–precision graph of IEEE Digital Library and our proposal system.

する必要がある．今回は第 1 著者の基準で文書の内容の評価を行い，正解集合を定めた．正解集合の大きさを**表 2** に示す．

4.1.2 実験結果と考察

前述した 3 つの問合せから，それぞれ次のようなキーワードを各検索システムに入力した．

(a) multimedia，document，database
(b) xml，retrieval
(c) multi，dimension，index

結果をそれぞれ**図 5**，**図 6** に示す．図中にある **ACM DL** は ACM Digital Library における検索結果であり，**IEEE DL** は IEEE Digital Library における検索結果である．再現率–適合率グラフは文献 3) の 3 章に記述されている 11 standard recall level に基づいた．また，各検索アルゴリズムを適用した結果，評価値が 0 となった文書はシステムが検索しなかったものとして検索結果に含めなかった．

検索結果を見ると，再現率が低い部分では ACM/IEEE Digital Library の方が本手法よりも高い場合があることが分かる．この原因として，tf/idf による重み付けを行う際の idf 値が正確な値ではなかったことがあげられる．これは IEEE Digital Library には大量の文書が収録されており，ある単語に対して適切な重みを計算することが容易であるのに対し，本システムに収録されている文書数は少ないため，適切に重みを計算できなかった点が原因であると考えられる．だが，再現率が高い部分ではすべての問合せで本研究の方が良い結果を得ることができた．

4.2 実 験 2

4.2.1 実 験 方 法

本研究で用いた多くの特徴量を使うことによって問合せの自由度を広げることで，より精度の高い検索を行うことができることを実験によって実証する．実験で扱うデータとして “2000 Digital Symposium Collection” に含まれる複数の会議録のうち，CoopIS, DASFAA, DOA, ICDE, KDD, PODS, SSDBM の 7 つの会議のものを使用した．PDF 文書数は 351 文書であり，1 つの論文が 1 つの PDF 文書に収録されている．以下に実験の方法について説明する．

(1) 利用者の問合せを考える．利用者は以前見たことのある文書を検索していると考え，次のような要素を用いた問合せを考える．
  (a) **文字+レイアウト**
  “multimedia” という文字列が紙面の左上にあるもの．
  (b) **文字+レイアウト，画像**
  “multimedia” という文字列が紙面の左上にあり，黒い画像があるもの．

(c) **文字，画像+レイアウト**
“multimedia” という文字列があり，右上に黒い画像があるもの．
(d) **文字+レイアウト，画像+レイアウト**
“multimedia” という文字列が左上にあり，右上に黒い画像があるもの．

つまり，利用者がどの程度の特徴量を問合せとして用いるかという観点から問合せの種類を決めた．

(2) 問合せに対してあらかじめ人手で正解集合を求めておく．
(3) 本研究で構築したシステムを用いて，2 種類の方法で答えを求める．
(a) 単語の頻度情報のみを用いた検索
(b) 問合せの要求に応じた特徴量を用いた検索
(4) あらかじめ求めておいた正解集合と比較することによって，再現率–適合率グラフを求める．

これらの問合せは，たとえば利用者がすでに見た論文を検索する場合などに用いられると考えられる．この場合には，利用者がどの程度該当する論文について記憶しているかによって問合せが決められる．本実験によって，記憶の度合いによる本システムの性能を示す．

実験 1 と同様に，本論文の第 1 著者の主観により正解集合を求めた．問合せ (a) の正解集合は 32 文書，問合せ (b) の正解集合は 19 文書，問合せ (c) の正解集合は 5 文書，そして問合せ (d) の正解集合は 2 文書であった．

4.2.2 実験結果と考察

本実験では自由度の異なる 2 つのシステムを比較するため，1 つの問合せから文字列だけの問合せ，利用者が指定したメディアを用いた問合せの 2 つを考えなければならない．そこで，基本的には利用者の問合せのうち文字列を指定した部分だけを用いて，文字列のみの問合せとした．つまりすべての問合せにおける文字列のみの問合せは，すべて「“multimedia” という文字列があるもの」という問合せになる．利用者が指定したメディアを用いた問合せについては，次のような問合せ拡張を行った．

問合せ (a)word multimedia **on** [0,0,1050,1500]
問合せ (b)word multimedia **on** [0,0,1050,1500] ,
image [1,0]
問合せ (c)word multimedia ,
image [1,0] **on** [1050,0,2100,1500]
問合せ (d)word multimedia **on** [0,0,1050,1500] ,
image [1,0] **on** [1050,0,2100,1500]

ここで，問合せとして「黒い画像」を指定している部分については，“**image** [1,0]” と拡張した．これは [1,0] の部分は完全に黒だけで構成された画像を表しており，黒に近い画像ほど評価値が高くなる．たとえば少し白が入った黒を利用者が表現するならば [0.9,0.1] などの指定を行うことも可能である．また，レイアウト情報の部分で “[0,0,1050,1500]” は左上を，“[1050,0,2100,1500]” は右上を表している．これは紙の大きさが 2100×3000 の大きさであることから計算してこれらの値を用いた．つまり，利用者は左上や右下などの指定ではなく，“[0,0,1000,1100]” などのように紙のある部分を指定することによってレイアウトを指定することができる．ほかに，GUI を用いた利用者の指定なども考えられる．

図 7 は，4.2.1 項で考えた問合せを入力し 3.4 節で求めた評価関数で得られた結果の再現率，適合率を表したものであり，*ARI* は相加平均，*GEO* は相乗平均，*HAR* は調和平均，*PRO* は PRO 関数，*MAX* は最大値，*MIN* は最小値，*normal* は単語の頻度情報だけを評価値にした場合の適合率，再現率を表している．また，問合せ (a) については 1 つのテキストに関する検索であり，複数のメディアに対する検索ではないため評価関数を用いない．そのため，本手法を用いた場合の再現率–適合率グラフは 1 つだけとなっている．その他の手法ではすべての評価関数を用いて評価を行った．実験 1 と同様，各検索アルゴリズムを適用した結果，評価値が 0 となった文書はシステムが検索しなかったものとして，検索結果に含めなかった．

実験結果から，レイアウト情報を用いた検索の場合は用いなかった場合と比較して適合率が向上したことが分かった．これは，レイアウト情報を問合せとして入力することが有効であったことを示している．だが，複数のメディアを用いた問合せが必ずしも有効でないことも分かった．たとえば問合せ (b) の場合を考えると，調和平均を評価関数として用いた場合の適合率よりもテキストのみを問合せとして用いた場合の適合率の方が高いことが分かる．よって，適切な評価関数を求めることは重要である．

また，複数の評価関数を用いて再現率と適合率の比較を行ったが，“PRO 関数” と呼ばれる評価関数を用いた場合に比較的高い値を示した．相加平均を用いた場合でもそれほど大きな適合率の低下は見られなかったが，相乗平均を用いた場合には再現率が低下した．これはテキストオブジェクトの評価値が 0 で画像オブジェクトの評価値が高い値であった場合，相乗平均は

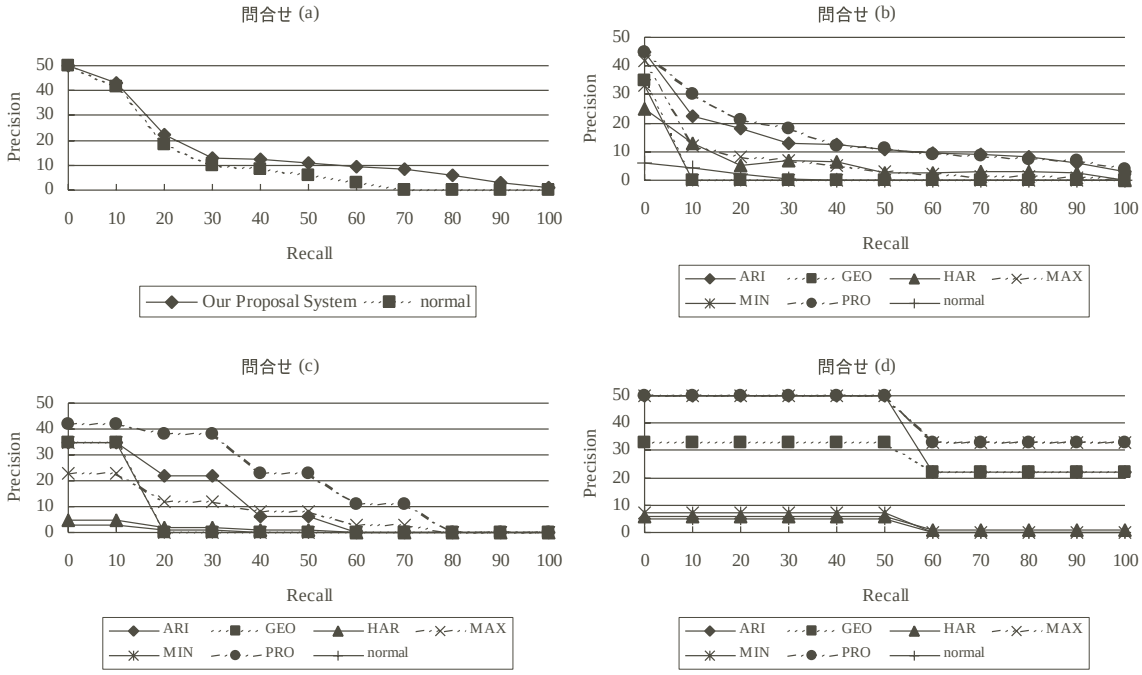

図 7 検索結果の再現率–適合率グラフによる比較
Fig. 7 Recall–precision graph of our proposal systems with six evaluation functions.

0 となってしまい，結果的には適合しないことになってしまうからであると考えられる．また，最大値を評価関数に用いると適合率が低下してしまった．これはある特定の部分が適合している場合に評価値が高くなってしまうために，文字列部分だけが適合して画像部分がまったく適合していなくても評価値が高くなってしまうからであると考えられる．つまり，各メディアの評価値すべてがある程度文書全体の評価値に反映された方が，あるメディアの評価値が文書全体の評価値に反映されるよりも良いことが分かった．

5. 結　　論

本論文では，複数のメディアから構成された電子文書を文字列だけでなく画像やレイアウトなどの特徴からも検索する方法について提案した．本手法の利点としては，以下のものがあげられる．

- 利用者が文書を検索する手がかりとして，単語の出現頻度情報だけではなく画像やレイアウトの情報を使うことで，より明確に利用者の興味を表現することができた．これは，複数の評価値を組み合わせる方法を本手法で提案することにより実現した．
- 電子文書の特徴量をベクトル化することによって，文書の適合度で順位付けを行うことができた．
- 複数の評価値を統合するための評価関数を比較することによって，最も良いと思われる評価関数が “PRO 関数” と呼ばれる関数であることが分かった．

本研究の今後の課題として以下のようなものがあげられる．

- 利用者が問合せに対して重み付けを行うことによって，より良い検索を行うことができると考えられるが，これらをふまえた問合せの拡張方法を考えなければならない．たとえば，利用者が “林檎の写真” という問合せを行った場合に，自動的に画像の特徴量として赤いものを指定し，画像に対する重みを大きくし，“林檎” という単語に対する重みを小さくするといったことが考えられる．これらを実現する方法として，多くのメディアを考慮したコーパスが必要であると考える．つまり，“林檎” → “赤い画像” といった変換が実現可能なコーパスを作成する手段を考えなければならない．
- 特徴量の種類として，さらに多くの特徴量を扱うことができると考えられるが，本研究で扱った特徴量以外の特徴量をベクトル化する方法，それらを評価する方法について考える．音声や映像に関する特徴量のベクトル化などが考えられる．

謝辞 本研究の一部は，文部科学省科学技術研究費（課題番号：11480088，12680417，12780309），ならびに科学技術振興事業団（JST）の戦略的基礎研究推進事業（CREST）「高度メディア社会の生活情報技術」プログラムの支援によるものである．ここに記して謝意を表します．

参 考 文 献

1) Adobe Systems Incorporated: *Postscript Language Reference Manual* (1985).
2) Adobe Systems Incorporated: *Portable Document Format Reference Manual* (1999).
3) Baeza-Yates, R. and Ribeiro-Neto, B.: *Modern Information Retrieval*, ACM Press (1999).
4) Sclaroff, S., Cascia, M.L., Taycher, L. and Sethi, S.: Unifying Texual and Visual Cues for Content-Based Image Retrieval on the World Wide Web, *Computer Vision and Image Understanding* (*CVIU*), Vol.75, No.1, pp.86–98 (1999).
5) Fujisawa, H., Shima, Y., Koga, M. and Murakami, T.: Automatically Organizing Document Bases Using Document Understanding Techniques, *Proc. 2nd Far-East Workshop on Future Database Systems*, pp.244–253 (1992).
6) Salton, G.: *Automatic Text Processing: The Transformational Analysis, and Retrieval of Information by Computer*, Addison-Wesley (1988).
7) Zhang, C., Meng, W., Zhang, Z. and Wu, Z.: WebSSQL—A Query Language for Multimedia Web Documents, *Proc. IEEE Advances in Digital Libraries 2000* (*ADL2000*), pp.58–67 (2000).
8) 石田和生，神谷俊之，市山俊治：既存文書のレイアウト情報付き構造化とその利用，技術報告 116–4–4, 情報処理学会研究報告，情報学基礎学研究会 (1996).
9) 串間和彦，赤間浩樹，紺谷精一，山室雅司：色や形状等の表層的特徴量にもとづく画像内容検索技術，情報処理学会論文誌，Vol.40, No.SIG3(TOD1), pp.171–184 (1999).

(平成 13 年 4 月 7 日受付)
(平成 13 年 7 月 5 日採録)

（担当編集委員 遠山 元道）

鈴木 優（学生会員）

1999 年神戸大学工学部情報知能工学科卒業．2001 年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了．同年奈良先端科学技術大学院大学情報科学研究科博士後期課程入学，現在に至る．マルチメディア情報検索に関する研究に従事．ACM 会員．

波多野賢治（正会員）

1995 年神戸大学工学部計測工学科卒業．1999 年同大学院自然科学研究科博士後期課程修了．博士（工学）．同年奈良先端科学技術大学院大学情報科学研究科助手，現在に至る．XML データベース，情報検索に関する研究に従事．ACM 会員．

吉川 正俊（正会員）

1980 年京都大学工学部情報工学科卒業．1985 年同大学院工学研究科博士後期課程修了．工学博士．同年京都産業大学計算機科学研究所講師．同大学工学部情報通信工学科助教授を経て，1993 年奈良先端科学技術大学院大学情報科学研究科助教授．2000 年国立情報学研究所ソフトウェア研究系客員助教授（併任）．1989〜1990 年南カリフォルニア大学客員研究員．1996〜1997 年ウォータルー大学客員准教授．XML データベース，多次元空間索引等の研究に従事．電子情報通信学会，ACM, IEEE Computer Society 各会員．

植村 俊亮（正会員）

1964 年京都大学工学部電子工学科卒業．1966 年同大学院工学研究科修士課程修了．同年通産省工業技術院電気試験所（現，電子技術総合研究所）入所．1988 年東京農工大学工学部数理情報工学科教授．1993 年奈良先端科学技術大学院大学情報科学研究科教授，現在に至る．工学博士．1970〜1971 年マサチューセッツ工科大学客員研究員．データベースシステム，自然言語処理，プログラム言語の研究に従事．電子情報通信学会，ACM, IEEE 等各会員．