

## マルチメディアデータベースにおける知識を用いた内容検索

岸田節子† 吉高淳夫† 吉光正典† 平川正人† 市川忠男†

広島大学大学院† 広島大学工学部†

オブジェクト指向マルチメディアデータベースにおいて、データの持つ意味内容に関する検索を行う手法について述べる。画像等のマルチメディアデータは、人間が見たり聞いたりしてはじめて認識できる意味内容を持っている。従来の内容検索手法では、この意味内容についての情報をテキスト等の形式であらかじめ付加しておき、それを用いて query を評価するものがほとんどであった。本手法では、データが含み得る内容についての知識を記述しておき、検索時にその知識を参照してデータから直接必要な情報を抽出し、query を評価する。この知識にはデータの処理方法等も記述しておくため、さまざまなメディアのデータを統一的に扱い、マルチメディアデータの持つ豊富な意味内容に関する検索を行うことができる。

## Content Retrieval Using Knowledge in a Multimedia Database

S.Kishida, A.Yoshitaka, M.Yoshimitsu, M.Hirakawa, T.Ichikawa

Faculty of Engineering, Hiroshima University  
Kagamiyama 1-4-1, Higashi-Hiroshima 724, Japan

This paper proposes content retrieval in an object-oriented multimedia database. For multimedia data, the content is recognized only when the user sees or listens to it. However, in most of the conventional approaches, additional information representing the meaning of each data item is specified prior to retrieval, and a query is evaluated by referring to the information. In our approach, knowledge about contents of multimedia data is provided to assist evaluation of queries. Here, the knowledge also includes methods of preprocessing data. Various media data can be managed uniformly and retrieval of rich contents in a multimedia data becomes achievable.

## 1 はじめに

近年、データベースの分野では、従来から扱われているテキストや数値データだけでなく、図形、画像、動画といったデータも一括して管理し、運用するためのマルチメディアデータベースの研究が注目を集めている。

このようにマルチメディアデータベースの研究が行われるようになった背景として、まず、光ディスクに代表される大容量の記憶装置、イメージスキャナや高解像度ディスプレイといったマルチメディアデータのための入出力装置等の飛躍的な進歩が挙げられる。

一方、データモデルの面では、マルチメディアデータを扱うためのシステムとして、現在最も普及しているリレーショナルデータモデルを中核とした試みや、オブジェクト指向データモデルに基づくもの等が提案されている [1]。特に、オブジェクト指向データモデルによるアプローチでは、文字、数字、図形、画像、音声、動画といった全く異質なデータを全てオブジェクトとみなすことによって、統一的に扱うことができるため、マルチメディアデータベースシステムに適したものの一つであると言える。

ところで、データベースのマルチメディア化が進む中で、マルチメディアデータの内容に関する検索を行うための手法の確立が重要になってきている。しかしながら、これまでのマルチメディアデータの内容検索は、各データにあらかじめ内容に関する属性情報をテキストや数値の形で付加しておき、検索時にはそれを評価に用いるという方法がほとんどであった。この方法では、全てのデータに対してあらかじめ人間が情報を付加しておかなければならないため、入力や更新時の負荷が大きく、更新時にデータの一貫性が失われやすい。また、あらか

じめ付加された情報についての検索のみしか行えないという問題点があった。

これに対して、データから内容に関する情報を直接抽出し、検索を行うアプローチも提案されている [5] [6] [7]。しかしこれらは、応用領域やデータの形式を限定し、それらに依存する性質を用いてデータから特徴量を抽出しているため、汎用性、移植性に乏しい。

そこで本研究では、知識を用いてマルチメディアデータの内容に関する検索を行う手法を提案する。この手法では、まず、内容検索の query を処理する方法や、データから内容に関する情報を抽出するための手続きを記述した知識を用意しておく。そして、検索時にこの知識を用いて query を評価することで、内容検索を行う。

2章でマルチメディアデータの内容検索について述べ、3章では内容検索のための知識の枠組みについて説明し、その知識を用いた内容検索処理について4章で説明する。5章で実際に作成したシステムについて説明し、最後に6章でまとめる。

## 2 マルチメディアデータの内容検索

### 2.1 従来のマルチメディアデータの内容検索

従来からデータベースで扱われてきた数値やテキストデータに対する検索は、データベースに蓄えられているデータそのものの値と query で示された検索条件とを直接比較して、query を評価するものであった。しかし、画像や音声といったマルチメディアデータに対しては、データそのものと検索条件とを直接比較してデータの意味内容に関する検索を行うことが困難である。これは、次のような理由からである。

- 数値やテキストデータは各データが単純値であるが、マルチメディアデータは複数の値がまとまって意味のあるデータを構成している。
- マルチメディアデータが持つ意味内容は、人間が見たり聞いたりしてはじめて認識できるもので、単純な値の比較だけではデータの持つ意味内容を判断できない。

そこで、この意味内容に関する検索を行うための手法が必要である。これまでに提案されているマルチメディアデータの内容検索方法は、次の2種類に分類できる。

1. あらかじめ各データに対して内容に関する情報をテキストや数値で記述しておき、検索時にはその値を用いて query を評価する [4]。
2. 応用領域やデータの形式を限定した上で、内容に関する情報をデータから自動的に抽出し、検索を行う [5] [6] [7]。

1の手法は実現が容易である反面、入力や更新時に負荷が大きく、更新時にデータの一貫性が失われやすい。また、マルチメディアデータが持つ豊富な意味内容を十分に生かした検索を行うことができない。

これに対して2の手法は、データから自動的に抽出された情報を用いて検索を行うものである。しかし従来の研究では、応用分野を限定した上で、その分野のデータの性質に基づいて特徴を抽出し、検索を行っている。例えば、グラフィックエディタを用いて描いた画像のみを扱う [5] [6] とか、商標パターンを扱うデータベースで各パターンから特徴を抽出して類似検索を行う [7] とかいったものである。これらの手法では、新たに異なるメディアのデータを扱おうとすると DBMS 自体を更新しなければならなくなり、汎用性に欠けている。

そこで本研究では、システムに内容検索のための知識を記述しておき、検索時にはその知識を用いて query を評価する手法を提案する。この知識の中には、データから内容検索に必要な情報を抽出する手続きや、query を評価できる形に変換する方法を記述しておき、さまざまな形式のデータを統一的に扱うことができるようにしている。また、知識の中にデータに対する操作を記述しておくため、メディア独立性 [3] を持ったシステムを構築することができる。

## 2.2 マルチメディアデータベースのためのデータモデル

マルチメディアデータベースを実現するためのデータモデルのアプローチとしては、主に次のようなものが挙げられる [3]。

1. 拡張リレーショナルデータモデルに基づくアプローチ
2. オブジェクト指向データモデルによるアプローチ

1は、広く普及しているリレーショナルデータベースにおいてマルチメディアデータを扱うために、抽象データ型を導入する等の拡張を行ったものである。

しかし、マルチメディアデータを扱うために適したデータモデルとして現在注目を集めているのは、2のアプローチである [1] [2]。これは、次のような理由からである。

- 異なるメディアのデータを全てオブジェクトとみなすことによって、それらを統一的に扱うことができる。
- ユーザに対して、メディアの違いを意識させることなく、統一したインタフェースを提供できる。

これらは、データと手続きのカプセル化、クラス階層に基づいたデータ構造や手続き

の継承、メッセージパッシングによる計算メカニズム、対象とするオブジェクトの種類によって振舞いを変えられるポリモアフィズムなど、オブジェクト指向が持つ機能によって支援されている。

そこで、本研究では、オブジェクト指向のアプローチに基づいて、マルチメディアデータベースを構築する。

### 2.3 オブジェクト指向データベースモデル MORE

我々はこれまでにオブジェクト指向データベースモデル MORE [8] を提案している。本研究では、この MORE に基づいてマルチメディアデータの内容検索を行えるようにしている。

MORE では、現実世界に存在する個々のもの(データ)をオブジェクトとし、共通の性質を持つオブジェクトの集合をクラスとして管理する。オブジェクトには、プリミティブオブジェクトと複合オブジェクトの2種類がある。プリミティブオブジェクト

は具体的な値を持つオブジェクトであり、複合オブジェクトはオブジェクトの集約操作により定義されるオブジェクトである。複合オブジェクトとそれを構成するオブジェクトとの集約関係を、アグリゲーション階層として管理する。また、クラス間の専化/汎化関係をクラス階層として管理する。

クラスの性質はクラス記述で定義される。また、クラスの性質はクラス階層に沿って継承される。

MORE のスキーマは、クラス階層とアグリゲーション階層によって構成される。このスキーマの例を Fig.1 に図示する。図中では、クラスを楕円で、オブジェクトを小さな丸で表している。楕円の包含関係でクラス階層を表し、オブジェクト(丸印)の間の矢印でアグリゲーション階層の参照関係を表している。

### 2.4 query の分類

本研究で扱う query の形式を次のように定める。

query : ( [path]attribute , operator , condition )

この形式に基づいて、query を次の2種類に分類する。

#### 1. 直接的な query

query で指定された attribute がスキーマ中に存在し、かつ、その attribute 値と condition の型が同じで比較可能なもの。

#### 2. 非直接的な query

query で指定された attribute がスキーマ中に存在しない、あるいは、存在してもその attribute 値と condition の型が違うために、そのままでは評価できないもの。

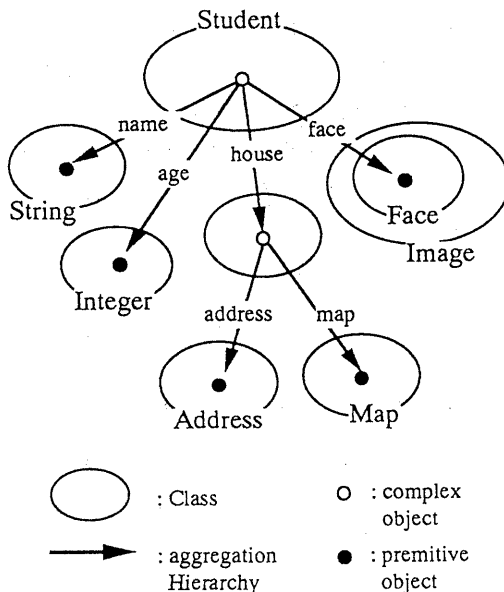


Fig.1 MORE スキーマ例

Fig.1に示したスキーマに基づいてデータが蓄えられているとして、次のような query を考える。

```
q1 : ( Student.age = 22 )
q2 : ( Student.hair = long )
q3 : ( Student.house is_next_to
      school )
```

q1 は、Student クラスに “age” という attribute が存在し、その値と “22” という condition とを “=” という operator を用いてそのまま比較することで処理できるので、直接的な query である。これに対して、q2 は、Student クラスに “hair” という attribute が不在のため、また q3 は、Student クラスに “house” という attribute は存在するが複合オブジェクトであるため、いずれもそのまま比較処理することができない。そのため、これらは非直接的な query である。

### 3 知識の表現

#### 3.1 内容検索概略

ここでは、本研究で提案するマルチメディアデータの content 検索手法について簡単に説明する。

マルチメディアデータは、人間が見たり聞いたりしてはじめて認識できる意味内容を持っている。本研究では、この意味内容のことを概念と呼ぶ。

まずこの概念を表現する手段として、Domain Knowledge と呼ぶ知識を導入する。具体的には、この知識には、

- 画像などのマルチメディアデータから、その概念に関する情報を検索時にいかに抽出するか。
- query で示された operator や condition を検索時にどう変換するか。

ということを記述しておく。

マルチメディアデータは多様であり、それに伴う操作もまた多様であるため、データの種類によって処理に必要な手続きは異なる。例えば、画像データであれば画像処理やパターン認識の技術を応用した手続きが必要になる。さらに、同じ画像データにおいても、応用領域や検索要求などの違いによって、効果的な特徴抽出処理方法も異なってくる。また、query で示される operator や condition も、データの種類によって、どのように解釈して検索できる形に変換すればよいかが異なる。そこで、上記の 2 点を知識の中に記述し、様々なデータに対する content 検索処理を可能にする。

また、仮想属性というものを導入し、Domain Knowledge によって既存の実データから抽出できる属性値を、あたかも実属性値として扱えるようにしている。そして検索のための query が提示された時、どの概念についての Domain Knowledge を参照すればよいか、ということ記述する。

仮想属性と Domain Knowledge を用いた非直接的な query に対する処理を先の q2 を用いておおまかに説明すると次のようになる (Fig.2)。まず、“hair” という仮想属性について、それを導出可能な実属性 “face” と対応づけし、実属性 “face” の値 (画像デー

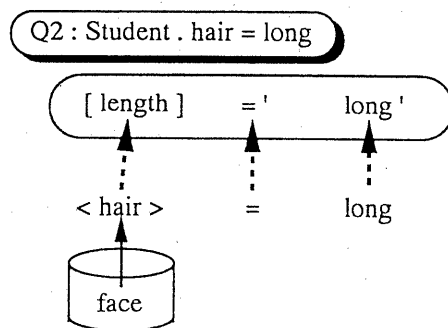


Fig.2 非直接的な query に対する処理

タ)から検索に必要な髪の長さの情報を抽出する。次に、query 中の “=” と “long” とを髪の長さについての比較ができる形に変換する。これらの抽出・変換過程において、Domain Knowledge を用いる。

### 3.2 Domain Knowledge

Domain Knowledge は、オブジェクト指向データベーススキーマ上のクラス記述の一項目であり、そのクラスのオブジェクトが含み得る概念について記述する。この Domain Knowledge の記述形式を Fig.3 に示す。Domain Knowledge は、3つの部分から構成されており、それぞれの役割を以下に説明する。

#### [FEATURE ITEM]

この概念が含み得る特徴を feature item として、その抽出手順と共に記述する。feature item は、検索時に各データから抽出する特徴量のための型のような役割をする。この型は数や文字といった単一値または構造を持ったもの等を取り得る。検索時に抽出手順にしたがって各データから抽出される特徴量は、この型に沿った値となる。

#### [OPERATOR]

query で用いられる operator が、実際にはどんな比較関数に対応しているかを feature item ごとに記述する。検索時には、この記述を参照して、operator を feature item の示すデータ型の値の下で比較できるように変換する。この比較関数は、各データから抽出された feature item と次に説明する condition が変換された値とを比較して、0.0 から 1.0 までの評価値を返す。評価値が大きいほど確からしいことを意味する。

(概念)

#### [FEATURE ITEM]

{ ( feature item , 抽出手順 ) , ... }

#### [OPERATOR]

{ ( operator , { ( feature item , 比較関数 ) , ... } ) , ... }

#### [CONDITION]

{ ( condition , feature item , value ) , ... }

Fig.3 Domain Knowledge の記述形式

#### [CONDITION]

query 中に現れる condition をどの feature item に対応づけするかを指定し、さらに condition が具体的にはどんな値と等価であるかを value に記述する。この value は feature item の示すデータ型の値を持つ。

Domain Knowledge の記述例を Fig.4 に示す。これは、顔写真のクラスの (hair) という概念について記述したものである。ここではまず、(hair) が持つ特徴として、[FEATURE ITEM] に “length” と “color” が、その抽出手順と共に記述されている。なお、“length”の型は1つの整数値、また“color”の型はRGBに対応する3つの整数値から構成されているとする。抽出手順は、ext.length\_proc および ext.color\_proc に記述されている。それらでは、まず顔写真の画像データに簡単な画像処理を施しておおまかに領域分割し、それらの領域の中で位置や大きさ等の制約を満たす領域を髪領域と判定した後で、その髪領域の縦方向の画素数を “length” とし、また髪領域の色を RGB 各色 256 段階で表した値を “color” に与える。

次に [OPERATOR] では、“=” について、長さに関する比較関数 cmp\_length\_func と色に関する比較関数 cmp\_color\_func が記述されている。これらの比較関数は、feature

```

(hair)
[ FEATURE ITEM ]
  { ( length , ext_length_proc ),
    ( color , ext_color_proc ) }
[ OPERATOR ]
  { ( = , { ( length , comp_length_func ),
            ( color , comp_color_func ) } ) }
[ CONDITION ]
  { ( long , length , 140      ),
    ( short , length , 0      ),
    ( black , color , [0 0 0] ),
    ( brown , color , [190 54 20] ) }

```

Fig.4 Domain Knowledge の記述例

itemとして各画像から抽出された値と condition から変換された値との距離を計算する。

[CONDITION]には、(hair)に関する query で取り得る4つの condition について記述している。この値は、feature item の示す型に沿って記述されるので、“long”や“short”は整数の単純値で、“black”や“brown”は3つの整数の組で表されている。例えば、“long”は“length”の値が140であり、“brown”は、“color”についてのRGBの値がそれぞれ190,54,20であることが定義されている。

### 3.3 仮想属性

仮想属性は、データベーススキーマ上のクラス記述の一項目として、実属性と同様、そのクラスの属性として定義したいクラスに記述する。この仮想属性は次の形式で記述する。

```

( 仮想属性名 , 参照するオブジェクトへのパス , Domain Knowledge 名 )

```

これは、非直接的な query が提示されたとき、その query 中にある attribute が、実

際にはどのクラスのオブジェクトが含むどんな概念に相当するか、ということを示すものである。これによって、実際には存在していない属性があたかも存在しているようにすることができる。

仮想属性の役割を Fig.5 を用いて説明する。この図では、Student クラスが“hair”という仮想属性を持っている。この“hair”は実際は Face クラスのオブジェクトの持つ (hair) という概念に相当するものであることが定義されているため、この属性が参照された時には、Face クラスの Domain Knowledge (hair) の記述にしたがって処理が行われる。

仮想属性は、非直接的な query を処理するために、データベーススキーマをどう捉えるかということを示している。一方 Domain Knowledge は、非直接的な query をデータの面からどう処理するかということを表している。この二つをそれぞれ別々に記述しておくことで、非直接的な query をスキーマ構造とデータとの両面から見て処理することができる。また Fig.6 に示すように、同じクラスのオブジェクトが別々のスキーマによって参照されている場合、その Domain Knowledge をそれぞれのスキーマの仮想属性から参照することも可能であるため、クラスの Domain Knowledge の再利用性も高めることができる。

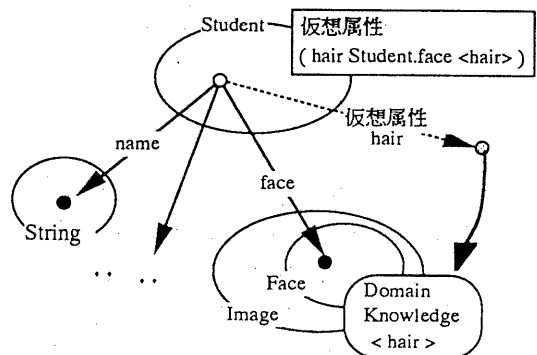


Fig.5 仮想属性の役割

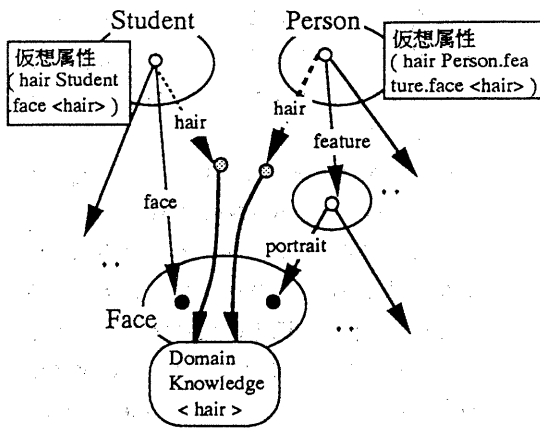


Fig.6 Domain Knowledge の再利用

#### 4 Domain Knowledge を用いた query 処理

query 処理の流れを Fig.7 に示す。各部分の処理を以下で説明する。

##### [query 解析]

提示された query の attribute とデータベーススキーマを見て、直接的な query か非直接的な query かを判断する。直接的な query ならば直接比較処理へ、非直接的な query ならば query 変換処理へ処理を移す。

##### [直接比較処理]

従来手法で、データベース内のデータと query の condition とを直接比較する。

##### [query 変換処理]

まず仮想属性を見て、query の attribute がどのクラスのどの概念に相当するか調べる。次に、指定された概念の Domain Knowledge を参照し、query の condition と operator を比較できる

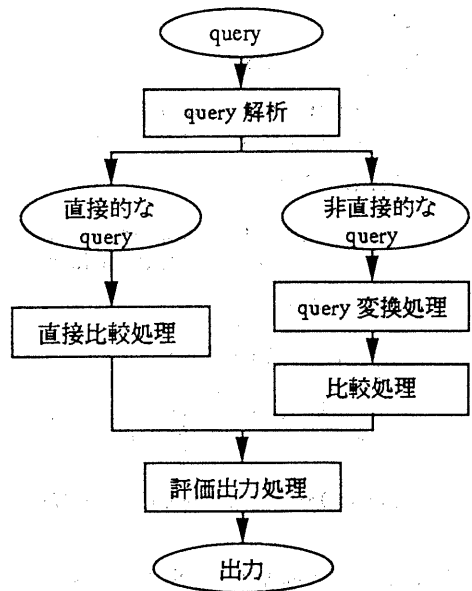


Fig.7 query 処理の流れ

形式 (feature item の型に沿った比較関数と値) にそれぞれ変換する。

##### [比較処理]

このクラスに属する各オブジェクトから、Domain Knowledge に記述してある feature item の抽出手順にしたがって feature item を抽出する。次に、condition の変換後の値と抽出された各 feature item とを、operator の変換によって得られた比較関数を用いて比較し、各オブジェクトに対して、0.0 から 1.0 までの評価値を返す。

##### [評価出力処理]

直接的な query に対しては、query を満たす全てのオブジェクトを出力する。非直接的な query に対しては、評価値の高いものから順に出力する。



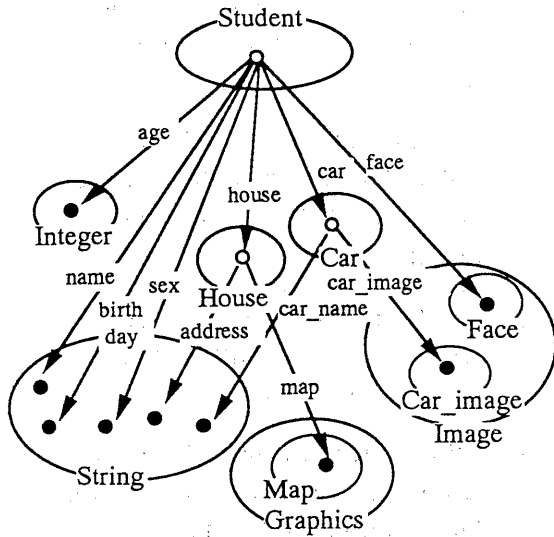


Fig.8 学生情報管理用スキーマ

## 5 インプリメンテーション

提案する手法の有効性を確認するために、オブジェクト指向マルチメディアデータベースシステムを実際に構築している。



Fig.9 顔写真のサンプルデータ

実験に用いたデータベーススキーマを Fig.8 に示す。これは、学生についての情報を蓄えるもので、学生の名前や年齢といった数値、テキストデータと共に、顔写真のような画像データも取り扱っている。

このスキーマ上の Face クラスに Fig.4 に示した Domain Knowledge を記述して、

q : ( Student.hair = long )

という非直接的な query を提示したところ、Fig.9 に示す顔写真データに対して、Fig.10 に示す髪の長さとお評価値が得られた。

実際の検索結果は、評価値が 0.0 でないものについて、その値が高い順に出力される。

## 6 おわりに

マルチメディアデータベースにおける、知識を用いた内容検索について述べた。この手法は、マルチメディアデータの内容に関する検索を行う際、各データにあらかじめ付加情報を持たせておくのではなく、知識を参照して検索時にデータから必要な情報を抽出しながら query を評価する。この知識の中には、データから内容検索に必要な

No.	1	2	3	4
length	0	134	65	75
評価値	0.00	0.90	0.00	0.00
No.	5	6	7	8
length	115	87	95	63
評価値	0.58	0.12	0.25	0.00

Fig.10 抽出された髪の長さとお評価値

な情報を抽出する手続きや、query を評価できる形に変換する方法を記述しておくため、様々な形式のデータを統一的に扱うことができる。

この手法を用いて内容検索を行うオブジェクト指向マルチメディアデータベースシステムを実際に作成し、画像データに対して内容検索実験を行ったところ、所望のデータが出力できた。なお、システムは、MS-WINDOWS 上で C 言語を用いて構築している。

今後は、内容検索実験を通して得られた知見を基に、システムの拡張を行う。具体的には、Domain Knowledge の記述において、Domain Knowledge の合成、参照、継承について考察し、様々な概念について柔軟に記述できるようにすること、ならびに query を文字や数値以外でも記述できるようにすることである。

#### 参考文献

- [1] 増永良文, “真のマルチメディアデータベースシステムの実現に向けて”, 電子情報通信学会論文誌, Vol.J74-D-I, No.8, pp447-449, 1991
- [2] 植村俊亮, “新世代データベースシステムに向けて”, 電子情報通信学会論文誌, Vol.J74-D-I, No.8, pp443-446, 1991
- [3] 小島功, 植村俊亮, “マルチメディアデータベースのためのデータモデリング”, 情報処理, Vol.28, No.6, pp685-693, 1987
- [4] S.Al-Hawamdeh, B.C.Ooi, R.Price, T.H.Tng L.Hui, “Nearest Neighbour Searching In A Picture Archive System”, International Conference On Multimedia Information Systems, 1991
- [5] P.Constantopoulos, J.Drakopoulos, Y.Yeorgaroudakis, “Retrieval Of Multimedia Documents By Pictorial Content: A Prototype System”, International Conference On Multimedia Information Systems, 1991
- [6] F.Rabitti, P.Stanchev, “GRIM\_DBMS: a GRaphical IMage DataBase Management System”, Visual Database Systems IFIP, 1989
- [7] 加藤俊一 他, “マルチメディア商標・意匠データベース TRADEMARK”, 信学技報, PRU88-9, 1988
- [8] K.Tsuda, K.Yamamoto, M.Hirakawa, M.Tanaka, T.Ichikawa, “MORE: An Object-Oriented Data Model with a Facility for Changing Object Structures”, IEEE Transactions on Knowledge and Data Engineering, Vol.3, No.4, pp444-460, 1991