4K−2

Query Model for Structured Objects *

Makoto ONIZUKA*　　Satoshi OKADA** †

* NTT Information and Communication Systems Laboratories,
** NTT Kansai Business Communications Headquarters

1 Introduction

Applications for retrieving still image data are becoming popular given the continued penetration of digital cameras. ORDBMSs and ODBMSs are typical of the software available to store and retrieve still images. Many recent DBMSs can calculate the similarity of an image to a key image based on several features. A feature is an attribute derived by transforming the original image through an image analysis algorithm, and is typically represented as a set of floating point numbers, often called a feature vector. For example, QBIC uses RGB histograms as one image feature([2]). One especially promising technique in content based retrieval extracts sub-images from image data, stores those sub-images and their parent images in a database, and locates the desired images according to the similarity between the sub-images and the key image([1][2]).

Nested relational quey models can be used to retrieve structured data. Unfortunately, its processing cost is high because we have to unnest the structured data, execute the desired operations (union, difference, selection, projection) on it, and renest the results to output the found images(middle part of Figure 1). On the other hand, object-oriented query models can handle structured objects by executing desired operations on several objects in a class and traversing to other objects at the database level, but application designers have to remove the duplication in the traversed objects at the application level(lower part of Figure 1).

The upper part of Figure 1 depicts the query process of our query model. In our query model, we can handle structured objects directly by assembling a set of classes and relations between classes as input and output units for query operations. It achieves faster query processing than the nested relational query models, because it doesn't have to process the nest and unnest operations. It is also faster than the object-oriented query model because application designers can depend on the database query optimizer for the whole of the retrieval process.

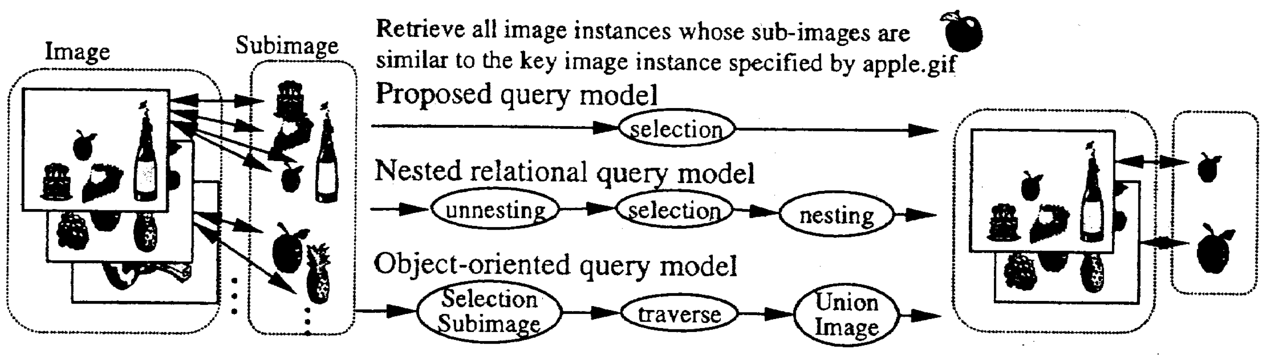

Figure 1: Image retrieval process

2 Query Model

Definition 2.1

$\mathcal{C}$: *Collection of all classes in a database.*
$\mathcal{A}$: *Collection of all attributes and methods in a database.*
$\mathcal{R}$: *Collection of all relations in a database.*
$\mathcal{I}$: *Collection of all instances in a database.*
$\mathcal{RI}$: *Collection of all relations_instances in a database.*
IWC: (C, I) *where* $C \in \mathcal{C}$, *and* $I \subset ext(C)$.
IWR: (R, RI) *where* $R \in \mathcal{R}$, *and* $RI \subset ext(R)$.
model: $(set(IWC), set(IWR))$.
$\mathcal{M}$: *Collection of model.*

*構造化オブジェクトに対する問い合わせモデル
†鬼塚 真（*NTT 情報通信研究所），岡田 敏（**NTT 関西法人営業本部）

Definition 2.2

Let $C, C_1, C_2 \in \mathcal{C}$, $I \subset ext(C), I_1 \subset ext(C_1)$, $I_2 \subset ext(C_2)$, $R_1, R_2 \in \mathcal{R}$, $RI_1 \subset ext(R_1)$, $RI_2 \subset ext(R_2)$, *and* $(IWCset, IWRset), (IWCset_1, IWRset_1), (IWCset_2, IWRset_2) \in \mathcal{M}$.

union

IWC **union:** $(C_1, I_1)\ union(C_1, I_2) = (C_1, I_1\ union\ I_2)$

IWR **union:** $(R_1, RI_1)\ union(R_1, RI_2) = (R_1, RI_1\ union\ RI_2)$

model **union:** $(IWCset_1, IWRset_1)\ union\ (IWCset_2, IWRset_2) =$
$(IWCset_1\ union\ IWCset_2, IWRset_1\ union\ IWRset_2)$

difference

IWC **difference:** $(C_1, I_1) - (C_1, I_2) = (C_1, I_1 - I_2)$

IWR **difference:** $(R_1, RI_1) - (R_1, RI_2) = (R_1, RI_1 - RI_2)$

model **difference:** $(IWCset_1, IWRset_1) - (IWCset_2, IWRset_2) =$
$(IWCset_1 - IWCset_2, IWRset_1 - IWRset_2)$

projection

IWC **projection:** *Let* $projections \subset \mathcal{A}$.

$\pi_{projections}(C, I) = (C', I')$

where $C'.attributes = (C.attributes\ intersect\ projections)\ union\ (C.method\ intersect\ projections)$, *and*

$I' = \{t\ |_{C'.attributes} |\ t \in I\}$

model **projection:** *Let* $IWCset = \{(C_1, I_1), ..., (C_n, I_n)\}$.

$\pi_{attributes}(IWCset, IWRset) = (\{\pi_{attributes}(C_1, I_1), ..., \pi_{attributes}(C_n, I_n)\}, IWRset)$

rename

IWC **rename:** $\rho_x(C, I) = (C', I)$ *where* $C'.name = x$

IWR **rename:**

$\rho_{from=x}(R, RI) = (R', RI)$ *where* $R'.from = x$

$\rho_{to=x}(R, RI) = (R', RI)$ *where* $R'.to = x$

model **rename:** *Let* $IWCset = \{(C_1, I_1), ..., (C_n, I_n)\}$, $IWRset = \{(R_1, RI_1), ..., (R_m, RI_m)\}$,
$\rho_x(C_i, I_i) = (C', I_i)(0 \le i \le n)$, *and* $R_l.from = C_i(0 \le l \le n), R_k.to = C_i(0 \le k \le n)$.

$\rho_{C_i=x}(IWCset, IWRset) =$
$(\{(C_1, I_1), ..., (C', I_i), ..., (C_n, I_n)\}, \{(R_1, RI_1), ..., \rho_{from=C'}(R_l, RI_l), ..., \rho_{to=C'}(R_k, RI_k), ..., (R_m, R_m)\})$

selection *Let* $IWCset = \{(C_1, I_1), ..., (C_n, I_n)\}$, *and* $IWRset = \{(R_1, RI_1), ..., (R_m, RI_m)\}$.

$\sigma_p(IWCset, IWRset) = (\ \{(C_1, I_1'), ..., (C_n, I_n')\},\ \{(R_1, RI_1'), ..., (R_m, RI_m')\})$

where $\forall i(0 \le i \le n), j(0 \le j \le m).I_i' \subset I_i, RI_j' \subset RI_j$, *and*

$\forall i_i, ri_j.i_i \in I_i', ri_j \in RI_j', p(i_1, ...i_n, ri_1, ...ri_m) = true$

For example, the query statement is expressed below according to the above query model definition.

SELECT x.data, y.data
FROM Image x, x.children y
WHERE y.features similar y.Key('apple.gif').features;

$\pi_{x.data,y.data}(\sigma_{y.features\ similar\ y.Key('apple.gif').features}(\rho_{x.children=y}(\rho_{Image=x}(IWCset, IWRset)))))$
where $IWCset = \{(Image, ext(Image)), (SubImage, ext(SubImage))\}$, *and*
$IWRset = \{(children, ext(children))\}$

3 Conclusion

In this paper, we proposed a query model that can handle structured objects directly by assembling a set of classes and relations as input and output units for query operations. Our query model is useful especially in applications that handle structured objects [1][2], because it offers faster query processing than other models, nested relational query models and object-oriented query models.

Our plan for future research is to extend our query model to handle the recursive structure of objects, query optimization, and hiearachical storage management which is suitable for structured image objects.

References

[1] M.Yamamuro et al, "ExSight - multimedia information retrieval system," Proc. Annual Pacific Telecommunications Conf., pp.734-739, 1998.

[2] M.Flickner et al, "Query by image and video content:The QBIC System," IEEE Com., vol.28, no.9, 1995.