Content-Initiated Organization of Mobile Image Repositories

Bo Yang and Ali R. Hurson
Dept. of Computer Science and Engineering, the Pennsylvania State University, University Park, USA
{byang, hurson}@cse.psu.edu

***Abstract*—Considerable research has been done on the content-based image delivery and access in distributed repositories. As noted in the literature, there is always a tradeoff between the image quality and the access speed. In addition, the overall performance is greatly determined by the distribution of the image data, specially, in a heterogeneous environment. In this paper, a semantic-based image access scheme for a distributed, mobile, heterogeneous database infrastructure, the Ubiquitous Content Summary Model, is presented that addresses both the data quality and performance issues. With the ability of summarizing the content information and guiding the data distribution, the proposed solution is distinguished by its mathematical representation and concise abstraction of the semantic contents of image data, which are further integrated to form a general overview of a image data source and its application of word relationships to construct a hierarchical meta-data based on the summary schemas allowing imprecise queries. Furthermore, it achieves the optimal performance in terms of searching cost. The fundamental structure of the proposed model is presented.**

***Index Terms*—Mobile Image Retrieval, Content Distribution, Data Integration, Ontology Model**

I. Introduction

SEARCHING and accessing image data from a collection of heterogeneous mobile data sources such as sensor or ad hoc networks is becoming important in many applications. Undoubtedly, image information is among the most powerful representations of the human thought ─ representation of entities as objects and representation of the complex objects in term of simpler objects [6]. However, image data is also one of the most non-manipulative structures in computers [2]. Indexing on images is rather difficult, which makes accessing or semantically organizing image data more difficult to realize. In a heterogeneous distributed environment, the autonomy and heterogeneity of local databases introduce additional complexity to efficient representation and manipulation of image data.

Manuscript received September 19, 2005. This work was supported in part by the Office of Naval Research under contract N00014-02-1-0282 and National Science Foundation under contract IIS-0324835.

Bo Yang is with the Pennsylvania State University, PA 16802-6106 USA (phone: 814-863-3646; e-mail: byang@cse.psu.edu).

Ali R. Hurson is with the Pennsylvania State University, PA 16802-6106 USA (phone: 814-865-9505; e-mail: hurson@cse.psu.edu).

Traditionally feature vectors as the representative of image data are employed to facilitate content-based query processing [3, 8, 17]. For an application-specific domain, the features from image data, empirically or heuristically, are extracted, integrated, and represented as some vectors according to the predefined application criteria. Due to the application-specific requirements, this approach lacks scalability, accuracy, robustness, and efficiency.

As a better alternative, one can manipulate the image data at a unified semantic-entity level. In this approach, the heterogeneous image data sources are integrated into a unified format — in spite of their differences. In addition, in this unified paradigm, different types of media (audio, video, image, and text) can be considered as inter-convertible objects. With the thorough understanding of the image content, this paradigm provides the QoS-guaranteed query processing with higher scalability and lower resource requirements.

In this work, we present an iterative method to represent a image object as a combinatorial expression of its simpler objects. In addition, a novel content-aware image indexing and accessing scheme for a heterogeneous distributed database environment is discussed. The proposed scheme, as a target platform, is used and enhanced based on the proposed content-aware accessing scheme ─ Ubiquitous Content Summary Model (UCSM). With the guidance of these summaries, the content-based image retrieval scheme offers superior performance than several well-known image indexing schemes, as demonstrated in our experiments.

This paper is organized into seven parts: Section 2 briefly overviews the related work and background materials. Section 3 addresses the concepts of the UCSM. Section 4 introduces the methodology framework. Section 5 analyzes the performance of the proposed model. Section 6 further discusses the description of image data contents within the framework of enhanced UCSM. Finally, section 7 draws the paper to a conclusion.

II. Background and Related Work

2.1 *Image Retrieval*

As witnessed by the literature [3-12], the research on the content-based image retrieval processing has focused on three interrelated issues:

- Representation of the image entities,
- Indexing and organization of the image entities, and
- Query processing strategies of image databases.

It should be noted that, most of the solutions that have advanced in the literature, study the image entities within the scope of the object oriented paradigm and hence, quantify a image data entity based on the features of its elementary objects.

Image data representation

The feature-based representation models can be further classified into four classes: the cluster-based organization, representative region approach, annotation-based organization, and decision tree-based organization [7-10].

The clustering-based approach partitions the image data objects into clusters of semantically similar objects [7, 20- 24]. The clustering approach can be further grouped into the supervised and unsupervised mode [7, 23, 24]. The supervised clustering approach utilizes the user's knowledge as input to cluster image objects, and hence it is not a general purpose clustering approach. As expected, the unsupervised clustering approach does not need the interaction with user. Hence, it is an ideal mechanism to cluster unknown image data automatically. Alternatively, the representative region approach, according to the Expectation Maximization (EM), constructs a simple description of the image objects based on several of the most representative regions of the objects [8, 25]. Motivated by the text attachment of image objects, the annotation-based organization paradigm makes use of manually or automatically added annotations [9]. Finally, integrated with the relevance feedback, the decision-tree-based approach organizes image data in a hierarchical fashion that separates the data by recursively applying decision rules [10, 26, 27, 28].

Efficient indexing scheme

Employment of efficient indexing is the key issue to the real-time retrieval of the image data . Efficient indexing is relatively more complicated when heterogeneous image data sources need to be integrated together. Two classes of indexing schemes have been discussed in the literature: The partition based indexing and the region-based indexing.

The partition-based indexing scheme, Quad-tree [11, 29], K-d-tree [15, 30], and VP-tree [16, 31], is a top-down process that recursively, divides the image object (or multidimensional feature space) into disjoint partitions while constructing a hierarchical data structure that represents the index of the image object. The region-based indexing scheme, R-tree [12], R*-tree [13], and SR-tree [14], takes a bottom-up approach in forming an access index ─ regard the image data as point objects in multi-dimensional feature space, employ some small regions to cover all the points, and then recursively, combines small regions into larger groups (how does the index form).

Efficient query processing

Searching for the image data is the crucial step in providing real-time image services. Based on the type of information submitted to the search engine, three searching strategies have been recognized: keyword querying, example matching, and fast browsing [18]. Cox et al. [18] proposed a Bayesian image retrieval system that accommodated all these three strategies.

Within the scope of a networked environment, the literature has addressed several practical image systems. The IBM Query by Image Content project (QBIC System by IBM Almaden Research Center) [3] allows users to query an image collection using features of image content – colors, textures, shapes, locations, and layout of images and image objects. Multi-dimensional feature vectors are employed to describe image content, with an R*-tree as the indexing structure. Speech Recognition (Jabber experimental system) [4] uses concept clustering based on indexing on audio content of a videoconference. It employs word recognition facility to set up an index based on the recognized words. To find the main topics and make a meaningful index, the Jabber system uses several lexical conglomerates, such as chains, trees, and clusters. The system uses surrounding words as restrictions, then compares the semantic distances of different relationships, and finally determines the relationship with minimal distance as the meaning of specified word. The Photobook System (developed by the MIT Media Lab) [5] is a system for Face recognition based on eigenvector descriptor. The Photobook System efficiently uses "distance-from-feature-space" (DFFS) to detect eigen-features. Given an input image, a feature distance-map is built by computing the DFFS at each pixel. The global minimum of this distance map is then selected as the best feature match.

In spite of the progress reported in the literature, the content-based image research is in its infancy . The scope of research in image database extends drastically when parameters such as autonomy, heterogeneity, mobility, and wireless limitation are added to the mix [6].

2.2 Multi-Database Systems

A multi-database system ***MDBS*** is a distributed system that acts as a global layer sitting on top of multiple preexisting distributed, autonomous, and heterogeneous local databases $\{\boldsymbol{LDBS_i}$, for $1 \leq i \leq n\}$ [1]. The local databases are connected via wired/wireless networks to form a global information sharing system. The local databases play dual roles in managing the data sources: On one hand, each local database $\boldsymbol{LDBS_i}$ located at site $\boldsymbol{LS_i}$, manages its local dataset $\boldsymbol{LDS_i}$. On the other hand, all local databases are harmonized under the restriction of a global access control mechanism. Figure 1 depicts a multi-database system.

Two types of requests exist in a multi-database system: local requests and global requests. The local requests are performed by the local database systems autonomously. The global requests, however, require the cooperation among local data-

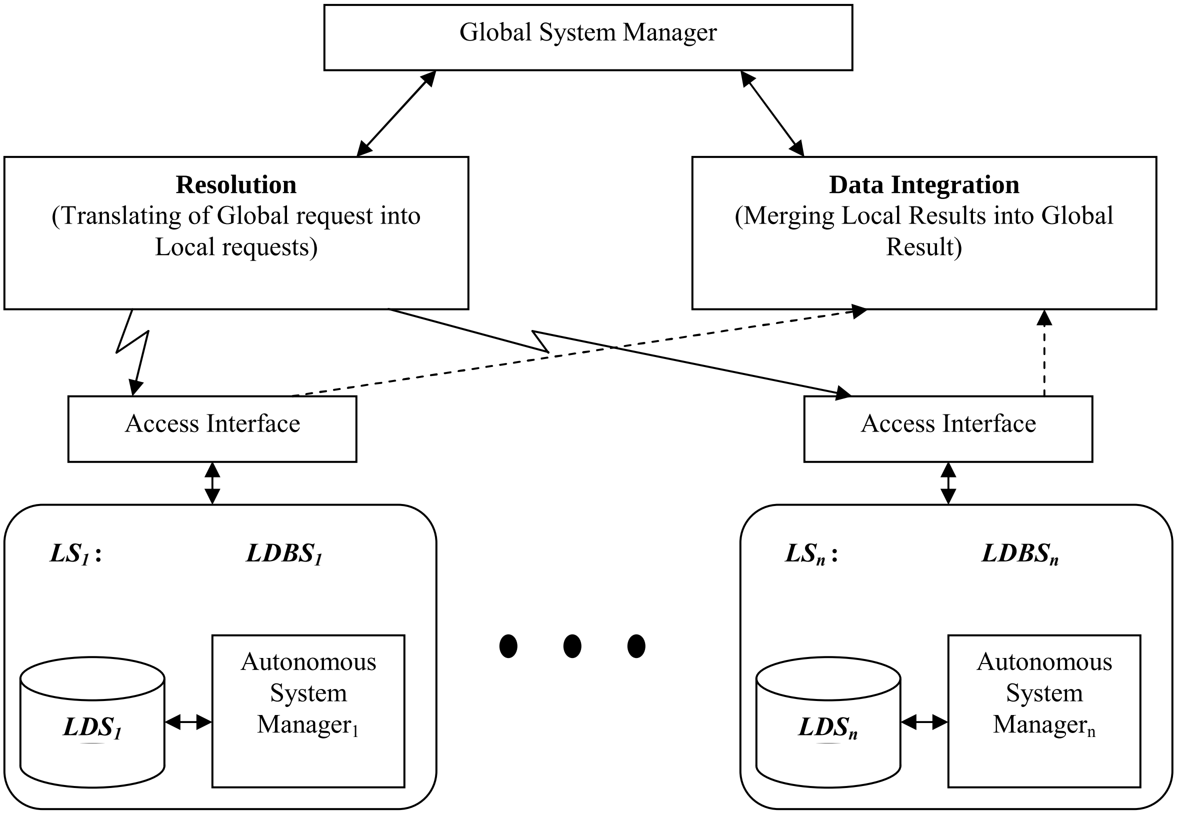

Figure 1: The Multi-database System.

bases. Normally, a global request $\boldsymbol{R}^g$ is the combination of a set of sub-requests $\{\boldsymbol{R}^g_{[\,i\,]}$, for $1 \le i \le m\}$, where each sub-request $\boldsymbol{R}^g_{[\,i\,]}$ is treated as a local request that can be executed on one of the local databases. The global request $\boldsymbol{R}^g$ is complete only after all the sub-requests are terminated at the local databases.

2.3 Summary Schemas Model

The Summary-Schemas Model (SSM) was proposed as a solution to large-scale multi-database systems [1, 19, 32]. It is a content-aware infrastructure that enables imprecise query processing on distributed heterogeneous data sources. The scalable content-aware query processing is made possible with the aid of an indexing meta-data based on the hierarchy of summary schemas, which comprises three major components: a thesaurus, a collection of autonomous local nodes, and a set of summary-schemas nodes.

The thesaurus provides an automated taxonomy that categorizes the local access terms and defines their semantic relationships ─ the thesaurus may utilize any of the off-the-shelf thesauruses (e.g. Roger's Thesaurus) as its basis. A Semantic-Distance Metric (SDM) is defined to provide quantitative measurement of "semantic similarity" between terms [19]. A local node is a physical database containing the data sources in different forms and representation, i.e., image data, textual data, formatted data. The local node is organized autonomously, on condition that its semantic content is communicated to the global mechanism at the thesaurus level. With the help of the thesaurus, the local access terms are classified, mapped, and integrated to their hypernyms. A summary schemas node is a virtual database concisely describing the semantic contents of its child (children) node(s). More detailed descriptions about the SSM can be found in [1, 19, 32].

In contrast with other multi-database solutions, the SSM has the following properties due to its unique semantic-based organization:

- The SSM allows automatic semantic-based data integration regardless of the heterogeneity of data sources.
- The SSM allows imprecise query processing. As a result, a user is able to submit his/her request in a free format notation.
- The SSM provides highly efficient content-based indexing capability.
- The SSM offers high scalability and robustness.

3 PRELIMINARIES

To overcome the aforementioned shortcomings of existing image systems, we introduced a novel image access paradigm based on the summary schemas model (SSM). As a scalable content-based scheme, the SSM prototype was originally proposed to resolve the name differences among semantically similar data in multi-database systems. Due to its concise structure and strong cross-modal representation capability, the SSM provides an efficient method of accessing image data.

3.1 Representation of Image Objects

The foundation of most image retrieval systems is the feature representation of image objects [2]. Extracted in the preprocessing stage of image representation, the features play an important role in quantizing the non-structured image data. The following definitions are the fundamental concepts of image retrieval systems.

Definition 1: Feature extraction

Assume $\boldsymbol{I} = \{I_j \mid 1 \le j \le n\}$ is a set of image objects, and $\boldsymbol{\Phi} = \{\varphi_i \mid 1 \le i \le m\}$ is the ordered mask of feature extraction priorities. The feature extraction process is a function $\boldsymbol{f}$: $\boldsymbol{I} \times \boldsymbol{\Phi} \to \boldsymbol{D}$, where $\boldsymbol{D}$ is the feature destination set. $\boldsymbol{D}$ could be a set of high-dimensional vectors, a set of cluster IDs, or real numbers indicating the semantic cluster that the image object belongs to.

In the image retrieval systems, there are two types of features: granule-level features and object-level features. The granule-level features are derived from the original format of image storage — i.e., those characteristics that directly or indirectly are obtained from the pixels, such as colors, textures, saturation. The object-level features, in contrast, are obtained from the recognition of the higher-level understanding of the image data — the semantic topics of the image data. In the aforementioned image retrieval system, the object-level features can be the recognized as elementary data items, shapes, spatial relationship, and etc.

Definition 2: Semantic distance

Suppose $\boldsymbol{I} = \{I_j \mid 1 \le j \le n\}$ is the set of image objects, and $\boldsymbol{\Phi} = \{\varphi_i \mid 1 \le i \le m\}$ is the ordered mask of feature extraction priorities. The semantic distance on feature φ_i is a function $\boldsymbol{g}^{\varphi i}$: $\boldsymbol{I} \times \boldsymbol{I} \to \boldsymbol{R}$, where $\boldsymbol{R}$ is the set of real numbers. The semantic distance function $\boldsymbol{g}^{\varphi i}$ compares two image objects and returns their semantic distance.

The function $\boldsymbol{g}^{\varphi i}$ satisfies the following characteristics:

1) *For any pair of image objects x and y:* $\mathbf{g}^{\varphi i}(x, y) \ge 0$,
2) $\mathbf{g}^{\varphi i}(x, y) = 0$ iff $x = y$,
3) *For any pair of image objects x and y:* $\mathbf{g}^{\varphi i}(x, y) = \mathbf{g}^{\varphi i}(y, x)$, *and*
4) *For image objects x, y, and z:* $\mathbf{g}^{\varphi i}(x, y) + \mathbf{g}^{\varphi i}(y, z) \le \mathbf{g}^{\varphi i}(x, z)$.

The semantic distance provides a quantized measure of comparing the difference between image objects. Based on the definition of semantic distance, we introduce the nearest neighbor concept that is widely used in most image retrieval systems.

Definition 3: The 1-nearest neighbor

Assume $\boldsymbol{I} = \{I_j \mid 1 \le j \le n\}$ is the set of image objects, $\boldsymbol{\Phi} = \{\varphi_i \mid 1 \le i \le m\}$ is the ordered mask of feature extraction priorities, $\boldsymbol{W} = \{\boldsymbol{w}_i \mid 1 \le i \le m\}$ is the set of weights of the feature extraction priorities, and $\boldsymbol{X}$ is the image object that is used as the query example. The nearest-neighbor searching process is a function $\boldsymbol{Q}$:

$$\boldsymbol{Q}(\boldsymbol{X}, \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{W}) = \{\boldsymbol{I_i} \mid \boldsymbol{I_i} = min\{\textstyle\sum_{k=1}^{m}(\boldsymbol{g}^{\varphi k}(\boldsymbol{X}, \boldsymbol{I}_j) * \boldsymbol{w}_k)\}_{j=1}^{n}\}$$

Definition 4: The K-nearest neighbor

Assume $\boldsymbol{I} = \{I_j \mid 1 \le j \le n\}$ is the set of image objects, $\boldsymbol{\Phi} = \{\varphi_i \mid 1 \le i \le m\}$ is the ordered mask of feature extraction priorities, $\boldsymbol{W} = \{\boldsymbol{w}_i \mid 1 \le i \le m\}$ is the set of weight of the feature extraction priorities, $\boldsymbol{K}$ is the parameter indicating the number of nearest neighbors, and $\boldsymbol{X}$ is the image object that is used as the query example. The K-nearest-neighbor searching process is a function $\boldsymbol{Q}^*$:

$$\boldsymbol{Q}^*(\boldsymbol{X}, \boldsymbol{K}, \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{W}) = \{\boldsymbol{I_i} \mid \ |\boldsymbol{Q}^*(\boldsymbol{X}, \boldsymbol{K}, \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{W})| = \boldsymbol{K}, \ \forall \boldsymbol{I}'_j \notin \boldsymbol{Q}^*(\boldsymbol{X}, \boldsymbol{K}, \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{W}), \ \textstyle\sum_{k=1}^{m}(\boldsymbol{g}^{\varphi k}(\boldsymbol{X}, \boldsymbol{I}_i) * \boldsymbol{w}_k) \le \sum_{k=1}^{m}(\boldsymbol{g}^{\varphi k}(\boldsymbol{X}, \boldsymbol{I}'_j) * \boldsymbol{w}_k)\}$$

The 1-nearest-neighbor search returns the image objects with the smallest semantic distance from the query example. The K-nearest-neighbor search returns K image objects, with the decreasing order of their similarities to the query example.

Based on the definition of semantic distance, the nearest-neighbor search can be performed in a multi-dimensional space of features. If we consider each feature as a dimension, the image object can be considered as a vertex in the multidimensional space of features. In this multidimensional space, the semantic distance between image objects is quantified as the spatial distance between vertices. The nearest neighbors should have similar positions as the querying example object. In another word, the nearest neighbors resides within a sphere whose center is the querying image object (Figure 2). In Figure 2, the semantic distance between any nearest neighbor and the querying image object is less than the radius of the sphere.

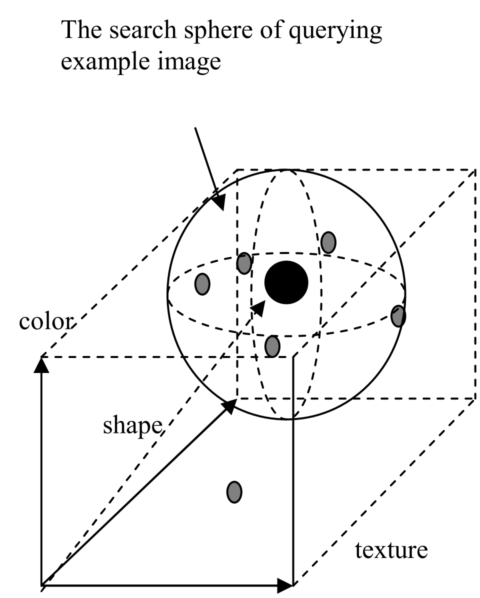

Figure 2: The search sphere for nearest neighbors.

Definition 5: The Elementary Entity

The elementary entities are those data entities that semantically represent basic objects (objects that cannot be divided further). Formally, the semantic contents of an elementary entity (E) can be considered as a first-order logic expression.

Let $E = f_1 \wedge f_2 \wedge \ldots \wedge f_n$, where $f_i = p_{i1} \vee p_{i2} \vee \ldots \vee p_{im}$ is the disjunction of some logic predicates (true/false values) and $p_{i1} \ldots p_{im}$ form a logic predicate set F_i, ─ In the feature-based image data sets, f_i indicates the i^{th} feature of the elementary entity. The semantic contents of an elementary entity can then be defined as:

$$E = \bigwedge_{i=1}^{n} (\bigvee_{j=1}^{m} p_{ij}), \qquad \text{for every } p_{ij} \in F_i.$$

Note that in any term $f_i = p_{i1} \vee p_{i2} \vee \ldots \vee p_{im}$, there is one and only one true predicate p_{ij}.. For instance, if $p_{i1}, p_{i2} \ldots p_{im}$ correspond to all possible color patterns, the semantic content of f_i at any time is a specific color pattern. Since f_i is disjunction of $p_{i1}, p_{i2} \ldots p_{im}$, the false predicates do not affect the final result. The content of an elementary entity is restricted by its conjunction terms $f_1, f_2 \ldots f_n$, which are the extracted features in application domains.

Definition 6: The Image Object

A image object is a collection of elementary entities. Given the above definition of elementary entities $E_1, E_2, \ldots, E_k$, the semantic contents of a image object can be defined as:

$$S = \bigcup_{i=1}^{k} E_j .$$

According to the definitions 5 and 6, a image object is considered as a combination of logic terms, whose value represents its semantic content. The analysis of semantic contents is then converted to the evaluation of logic terms and their combinations. This content representation approach offers the following advantages:

- The logic terms provide a convenient way to describe semantic contents concisely and precisely ─ The features of any elementary image object can be determined easily, hence, the semantic content of a complex image object can also be obtained using logic computations. The similarity between objects can be considered as the equivalence of their corresponding logic terms.
- This logic representation of image content is often more space efficient than feature vector. In a specific image database system, the feature vector is often of fixed length to facilitate the operations. However, for some objects, some features may be null. Although these null features do not contribute to the semantic contents of image objects, they still occupy space in the feature vectors, and hence lower space utilization. In contrast, the logic representation can improve storage utilization by eliminating the null features from logic terms.
- Compared with feature vectors, the logic terms provide an understanding of image contents that is closer to human perception.
- Optimization can be easily performed on logic terms using mathematical analysis. By replacing long terms with mathematically equivalent terms of shorter lengths, the image representation can be automatically optimized.
- Based on the equivalence of logic terms, the semantically similar objects can be easily found and grouped into the same clusters. This organization facilitates the nearest-neighbor retrieval, and at the same time reduces overlapping and redundancy. Hence, searching efficiency and storage utilization are both improved.

An interesting issue is that the logic representation approach can be seamlessly integrated with SSM. Each concept in the logic representation can find its counterpart in SSM, and hence can be performed within the domain of SSM. For instance, the equivalence between logic terms can be considered as the synonym relationship between summary schemas. Hence, the operation of finding equivalent terms in logic domain can be mapped to searching for synonyms in summary schemas domain. Similarly, other relationships between logic terms can also be conveniently represented in SSM. If term A is equal to a part of term B, then this "inclusion" relationship between A and B can be described with hypernym and hyponym relationships in SSM. Considering the strengths of SSM in organizing data [1, 19], we incorporate the logic representation within the framework of SSM. For simplicity, we consider logic term and summary schema as the same concept in the remaining part of this paper.

3.2 The Rationale of UCSM

The UCSM, the integration of SSM with logic predicate representation of image data contents, takes the aforementioned concepts further to a more general representation of image contents, with a series of interrelated access terms representing the relationship among image objects at the data base granularity level. As noted before, the SSM organizes and classify the data sources based on database semantics – database schemas and summary schemas, and word relationships. Within the SSM terminology then, a database schema is a group of access terms that describe the contents of image data source. A summary schema is a concise abstract of semantic contents of a group of database schemas. The summary schemas are connected through synonym, hypernym, and hyponym links. These links are logically used to represent the semantic relationship among image data objects.

Synonym links in the UCSM hierarchy are used to represent the semantically similar data entities regardless of their representation and/or access term differences. Refer to our methodology; a synonym link represents equivalence relationship between two image (or two groups of image) data objects. For instance, assume in an image environment, photos are grouped according to their authors. As a result, two similar photos taken by different authors will be kept in different databases. To represent similarity relationship between these two photos kept in two different databases, the UCSM employs synonym links to connect and group the similar image objects together.

- A hypernym is the generalized description of the common characteristics of a group of data entities. For instance, the hypernym of dogs, monkeys, and horses is mammal. To find the proper hypernyms of image objects, the UCSM maintains an on-line system taxonomy that provides the mapping from image objects to hypernym terms. Based on the hypernyms of image objects, the UCSM can generate the higher-level hypernyms that describe the more comprehensive concepts. For example, the hypernym of mammals, birds, fish and reptiles is animal. Recursive application of hypernym relation generates the hierarchical meta-data of the UCSM. This in tern conceptually gives a concise semantic view of all the globally shared image objects. Refer to our methodology; a hypernym link represents "a member of" relationship.

- A hyponym is the counter concept of a hypernym. It is the specialized description of the precise characteristics of image objects. It inherits the abstract description from its direct hypernym, and possesses its own particular features. The UCSM uses hyponyms links to indicate the hyponyms of every hypernym. These links compose the routes from the most abstract descriptions to the specific image objects.

One of the merits of the UCSM is the ease of nearest-neighbor search operation. In the UCSM, the nearest neighbors are considered as synonyms that are connected through synonym links. As a result, the nearest-neighbor search is simplified into a process of finding the synonym links. In other indexing models, the nearest neighbor indexing is a time-consuming process that requires searching through a subset of the distributed image databases [13, 14, 16].

3.3 The Structure of Summaries

The heart of the UCSM prototype is the generation of summary schemas, which imply the semantic content of image objects. Motivated by the observation that the low accuracy of the present image retrieval systems is due to the improper selection of granule-level features as the representation foundation, UCSM prototype employs the object-level features obtained from some computer vision algorithms [17, 25]. Since these object-level features usually have stronger descriptive capability than granule-level features, the summary schemas are able to describe the semantic contents of image data using more concise terms.

To represent the content of image objects in a computer-friendly structural fashion, the UCSM organizes the image objects into layers according to their semantic contents. A image object, say, an image, can be considered as the combination of a series of elementary entities, such as animals, vehicles, and buildings.

4 Proposed Methodology

Based on the summary schemas topology, the image databases are organized in a hierarchy, which consists of leaf nodes and intermediate summary schema nodes. The leaf nodes, containing the real data, are clustered according to their semantic contents. The common information of each group is extracted and kept in a higher-level summary schema node. This semantic summarization process continues until it reaches the root node. Consequently, the root node keeps the most abstract view of all the globally shared data objects. Traversal from root node to leaf nodes, the UCSM hierarchy provides a gradually refining method to find the image objects.

This UCSM hierarchy is notable in its strong support to content-based image retrieval. The query can be submitted at any summary schema node as well as the local databases. The query is resolved as a series of matching query requirements with summary schemas. The query processor first compares the summary schema's entry at the query origin node with the query goal. In case of a successful match, then the query processor returns the accessing terms as the result, if the query is originated at a leaf node, or query goal is sent to the proper child (children) node (s). Otherwise, the query goal moves up the summary schema's hierarchy and tries to match the query at a higher level. This process continues until the query goal reaches leaf node (s) or the query goal reaches root without a successful match. Based on our experimental result [32], the height of the UCSM hierarchy is short, which by default implies efficient search process.

In the UCSM, a user could issue either imprecise or precise queries. Imprecise queries are those that may have access terms different from local access terms and/or may not specify any location of the data. Precise queries, on the other hand, use exact local access terms and also give specific data location. A precise query can be resolved by sending the query directly to the specified database whereas the process of resolving an imprecise query is more involved in identifying semantic intents of the user's query and then, based on that intention, the query shall be resolved.

The hierarchical structure of the UCSM is used to resolve imprecise queries. The query resolution starts at the node issuing the query. Each term 'a' in the query is compared with the terms in the schema 's' at that node. If the SDM between all query terms and schema terms is less than or equal to some specified threshold SDM, the query is resolved either at that node (if it is a local node) or at the children of that node (if it is an SSM node). On the other hand, if 'a' and 's' are not linguistically related; hence, not matched, the search proceeds to the parent of the current node. This process will recursively continue until either the search reaches the top of the UCSM hierarchy and fails with no possible downward search, or the search fails at a particular node on a downward traversal, or the search reaches a local-node where the query is resolved. The search fails at a specific node when the query terms do not match the schema terms at that node.

Given a random set of image objects in a heterogeneous multi-database environment, the UCSM prototype relies on its summarization capability to construct a hierarchical indexing structure for these objects. Hence, finding proper content integration methods is the crucial step to show the effectiveness of the UCSM. Two classes of content integration are employed in the UCSM framework:

- Replacing a set of specific terms with a more general term (hypernym relation), such as summarizing "car", "bus", and "truck" into a more abstract concept "auto"; and

- Reorganizing combinations of features to a more concise description, such as changing {[(object = dog) $\wedge$ (color = grey)] $\bigcup$ [(object = dog) $\wedge$ (color = white)]} into a shorter equivalent term {(object = dog) $\wedge$ [(color = grey) $\vee$ (color = white)]}.

The first type of content integration is automated and relies on a system thesaurus [19, 32]. The second type, however, is an intriguing new issue that has not been explored. This content integration process, if resolved with properly designed strategies, would drastically reduce the cost of content-based retrieval in image databases.

Our goal in the content integration process is to specify the hidden semantic relationships among the image objects using an effective analytical comparison of the features. Inspired by the formation of Karnaugh Maps, we designed a combinatorial optimization table to shorten the complex combinations of features into condensed logic terms.

A UCSM-based indexing hierarchy is constructed during this content integration process. Compared with other indexing models, the UCSM hierarchy provides a more efficient content description by exploiting the unique summary representation of image objects. Our experimental results show that the UCSM has superior performance than some classic image indexing models, such as R*-tree and M-tree.

5 Theoretical Study

In this section, the performance of the proposed UCSM-based searching scheme is analyzed. As it is expected an effective content-based retrieval mechanism requires the ability to capture the semantic contents of the data objects accurately and efficient data searching. We analyze the performance of the UCSM based on two performance metrics; the size of the summary schemas and the searching cost in the summary-schemas hierarchy. Some presumptions are given to simplify the analysis process and final conclusions. The rationality of performance analysis is further supported by our simulation results.

5.1 The Analysis of Summary Schemas

In section 4, the semantic contents of image objects were mapped to a multidimensional space of features, then expressed as the disjunction of some first-order logic terms, and finally converted to a concise representation with the help of a combinatory optimization table. We now justify the rationality of summary schemas by showing that the size of the summary schemas is drastically shortened after optimization.

The size of summary schemas is measured by the number of predicates, which is comparable with the number of features in most of the other content-based indexing models. Reducing the number of predicates can reduce the number of comparisons in image object matching and consequently the communication cost during the query processing.

We assume a image object (say, an image) I having K elementary entities $E_1, E_2, \ldots, E_k$. Each elementary entity is within the multidimensional feature space indicated by $f_1, f_2, \ldots, f_n$, where $f_i = p_{i1} \vee p_{i2} \vee \ldots \vee p_{im}$ is the disjunction of some logic predicates. As mentioned in section 2, the semantic content of the image object I can be represented as the union of the elementary entities, which are expressed as the conjunctions of predicates. Refer to *Definitions 5* and *6*, we have the following expression of semantic content:

$$S = \bigcup_{i=1}^{k} E_i = \bigcup_{i=1}^{k} [\bigwedge_{j=1}^{n} (\bigvee_{h=1}^{m} p_{jh})], \text{ for every } p_{jh} \in F_j$$

Since the semantic content of feature f_i is uniquely determined by the true predicate p_{ix} within $p_{i1}, p_{i2}, \ldots, p_{im}$, we change the above equation into a simpler form:

$$S = \bigcup_{i=1}^{k} (\bigwedge_{j=1}^{n} p_j^{(i)})$$

where $p_j^{(i)}$ is the true predicate of the *jth* feature of the *ith* elementary entity.

Let S^* be the final result from the combinatory optimization table. Given the definition of combinatory optimization table, S^* by default expresses the same semantic content as S. According to step 4 of the optimization method, S^* is the union of a collection of clusters $C_1, C_2, \ldots, C_q$, with each cluster indicating several elementary entities. Hence, S^* can be expressed as the following:

$$S^* = \bigcup_{i=1}^{q} C_i.$$

As mentioned earlier, each cluster corresponds to a rectangular region in the combinatory optimization table. Assume cluster C_i is horizontally indicated by labels $L_1', L_2', \ldots, L_r'$, and vertically indicated by labels $L_1'', L_2'', \ldots, L_s''$. Here any label in $L_1', L_2', \ldots, L_r'$ or $L_1'', L_2'', \ldots, L_s''$ can be the conjunction of several predicates in equation (2). For instance, L_1' may be (object = cat) $\wedge$ (color = grey). Then C_i can be expressed as $(L_1' \vee L_2' \vee \ldots \vee L_r') \wedge (L_1'' \vee L_2'' \vee \ldots \vee L_s'')$, or

$$\bigvee_{i=1}^{r} [\bigvee_{j=1}^{s} (L_i' \wedge L_j'')].$$

When representing the clusters with labels, if a cluster is a whole row/column, then the label for the row/column can be omitted in the representation. For instance, if all texture patterns are in a cluster, then this cluster does not need the feature "texture" in its representation. For the clusters that do not contain whole rows/columns, avoiding overlapping with other clusters can reduce the size of summary schemas.

5.2 The Search Cost

Some content-based indexing models evaluated searching cost in terms of the number of comparisons [18], while others use the number of disk accesses as the searching cost [14, 16]. We believe that both parameters should be accounted when determining the searching cost. In this section, the searching cost of the summary-schemas hierarchy is calculated as the average number of accesses at the summary-schemas nodes (number of comparisons) and local nodes (number of disk accesses).

We assume a set of n image objects, $I_1, I_2, \ldots, I_n$ and the following notations in our analysis:

- $P(I_i)$: The probability of being queried for image object I_i.
- $\overline{W}$: The average searching cost for all image objects in any indexing tree model.
- $W(I_i)$: The searching cost for image object I_i in any indexing tree model.
- $N(I_i)$: The number of nodes on the path from root node to image object I_i in any indexing tree model.
- $\overline{W}^*$: The average searching cost for all image objects in summary schemas model.
- $W^*(I_i)$: The searching cost for image object I_i in summary schemas model.
- $N^*(I_i)$: The number of nodes on the path from root node to image object I_i in summary schemas model.

Given the above notations, the searching cost for a request composed on n random objects is:

$$\overline{W} = \sum_{i=1}^{n} [P(I_i)\, W(I_i)] \tag{1}$$

Considering the definitions of the indexing models [11-18], the content-based searching always starts from the root node, traverses within the indexing tree, and finally arrives at the image object I_i. Thus,

$$W(I_i) \geq N(I_i) \tag{2}$$

$$\overline{W} = \sum_{i=1}^{n} [P(I_i)\, W(I_i)] \geq \sum_{i=1}^{n} [P(I_i)\, N(I_i)] \tag{3}$$

Lemma 1: The UCSM hierarchy does not contain any form of overlapping between its branches.

The elimination of overlapping between branches of the UCSM hierarchy is due to the existence of synonym links. While the other indexing models (R-tree family, SS-tree, etc.) are striving for the reduction of overlapping, the UCSM hierarchy can completely remove the overlapping data by adding some synonym links to other branches.

Proposition 1: Given a fixed set of image objects, the UCSM hierarchy has less or equal height than any indexing tree.

Proof. We will prove that any indexing tree can be described using the UCSM hierarchy with less or equal height. Given any arbitrary set of image objects $\boldsymbol{I} = \{I_1, I_2, \ldots, I_n\}$ and any indexing tree model $\boldsymbol{M}$, we can construct an equivalent UCSM hierarchy in the following way:

Let $\boldsymbol{T}$ be the indexing tree generated from applying indexing model $\boldsymbol{M}$ to the image data set $\boldsymbol{I}$. And for any node $\boldsymbol{n_i}$ in tree $\boldsymbol{T}$, let ***feature*** ($\boldsymbol{n_i}$) be the set of features that globally identify node $\boldsymbol{n_i}$, ***parent*** ($\boldsymbol{n_i}$) denote the parent node of $\boldsymbol{n_i}$, and ***children*** ($\boldsymbol{n_i}$) be the set of child (children) node(s) of $\boldsymbol{n_i}$.

First, we group the leaf nodes into clusters $\boldsymbol{C_1}, \boldsymbol{C_2}, \ldots, \boldsymbol{C_k}$ according to common parents. For any cluster $\boldsymbol{C_j}$, make a union of all features of the nodes in this cluster to get the features for the common parent node. That is to say, suppose $\boldsymbol{n^*}$ is the common parent node of cluster $\boldsymbol{C_j}$, ***feature*** ($\boldsymbol{n^*}$) = $\bigcup_{ni \in Cj}$ ***feature*** ($\boldsymbol{n_i}$). The rationale behind this union is the fact that any node in the tree-based indexing structure can be identified by the route from the root to that node, which can also be determined by the features available at that node.

Next, we can use the aforementioned summarization process to generate a proper summary schema for the parent node $\boldsymbol{n^*}$. By recursively making abstraction, we construct a UCSM hierarchy with no more height than the indexing tree $\boldsymbol{T}$. According to *Lemma 1*, this UCSM hierarchy does not contain any overlapping, which may further reduce the height of the UCSM hierarchy. Hence, the UCSM hierarchy can describe any feature-based indexing tree with less or equal height. Or in another word, for any image object I_i, we have $N^*(I_i) \leq N(I_i)$.

As mentioned earlier in sections 3 and 4, the query can be submitted at any arbitrary summary-schemas node. In particular, when a K-nearest-neighbor query is submitted to the summary-schemas model, the searching is restricted within a small region rather than the whole indexing hierarchy. Assume the nearest neighbors are ordered by their similarities as $I'_1, I'_2, \ldots, I'_K$, the searching of I'_2 will be restricted within an area near the place of I'_1, which makes $W^*(I'_2) \leq N^*(I'_2)$. Hence,

$$\overline{W}^* = \sum_{i=1}^{n} [P(I_i)\, W^*(I'_i)] \leq \sum_{i=1}^{n} [P(I_i)\, N^*(I'_i)] \tag{4}$$

Considering equation (3) and *Proposition 1*, we obtain

$$\overline{W}^* \leq \sum_{i=1}^{n} [P(I_i)\, N^*(I'_i)] \leq \sum_{i=1}^{n} [P(I_i)\, N(I'_i)] \leq \overline{W} \tag{5}$$

Hence, the UCSM achieves the optimal performance in terms of searching cost.

6 FURTHER DISCUSSIONS

In addition to the performance consideration, another important factor – imprecise query processing – favors the choice of summary-schemas model as the underlying platform for content-based indexing. Most of the previous researches [11-18] in content-based retrieval focus on searching cost and similarity comparisons, and do not consider the imprecise query processing. As mentioned earlier, the summary-schemas hierarchy contains two types of summary schemas: the lower-level summary schemas generated by optimization of features, and the higher-level summary schemas constructed from content abstraction of lower-level summary schemas. The higher-level summary schemas may reveal some semantic content beyond the features extracted from the underlying data objects. For example, an image containing “flowers” and “smiling faces” may express the concept of “happiness”. For simplicity, we denote the lower-level summary schemas as “quantitative summaries”, and denote the higher-level summary schemas as “descriptive summaries”.

Section 4 presented an optimization algorithm for generating quantitative summaries. However, these quantitative summaries may not be able to reveal the implication of image objects. For instances, gestures, facial expressions, and background settings may have some implications that can only be extracted with human senses. Fortunately, these implications can be integrated within the higher-level summary schemas (descriptive summaries).

The descriptive summaries obtain the implications with the help of some common-sense rules, which indicate the semantic relationships between visual components and their symbolic meanings. For instance, "sun + flowers + smile" means "happiness", and "white doves + olive" symbolizes "peace". Some complex image objects may generate multi-level descriptive summaries.

With descriptive summaries, an imprecise query can be processed as follows: First, find a summary schema that matches with the query; then decompose the imprecise query into simpler descriptive summaries (or quantitative summaries) as sub queries; and finally combine the results from the decomposed sub queries. The capability of processing imprecise queries drastically enhances the searching power of the UCSM-based search engine, and makes the UCSM distinguished from other content-based indexing models.

7 CONCLUSIONS

We proposed a novel content-aware retrieval model for image data objects in heterogeneous distributed database environment. In contrast with the traditional feature-based indexing models, the proposed model employs a concise descriptive term – ubiquitous content summary – to represent the semantic contents of image objects. In short, the proposed model offers the following advantages: (1) the concise summary accurately represents the semantic contents of image objects using optimized logic terms; (2) the descriptive summary enables the search engine with capability of handling imprecise queries; and (3) the performance of content-based indexing within the UCSM hierarchy is optimal in terms of searching cost. Our future work would include improvements of the UCSM prototype, such as more efficient summarization strategies and adaptation to wireless network environments.

REFERENCES

[1] A. R. Hurson and M. W. Bright. Multidatabase Systems: An Advanced Concept in Handling Distributed Data. 1991, Advances in Computers 32: 149-200.

[2] W. I. Grosky. Managing image information in database systems. Communications of the ACM, 1997, 40 (12): 73-80.

[3] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yahker. Query by image and video content: The QBIC system. IEEE Computer, 1995, 28 (9): 23-32.

[4] J. Kominek and R. Kazman. Accessing image through concept clustering. Proceedings of the SIGCHI conference on human factors in computing systems, 1997: 19-26.

[5] A. P. Pentland, B. Moghaddam, T. Starner, M. Turk. View-based and Modular Eigenspaces for Face Recognition, IEEE Conf. on Computer Vision and Pattern Recognition, 1994: 84-91.

[6] G. Auffret, J. Foote, C. Li, B. Shahraray, T. Syeda-Mahmood, and H. Zhang. Image access and retrieval (panel session): The state of the art and future directions, Proceedings of the seventh ACM international conference on image, 1999, ACM Image (1): 443-445.

[7] J. B. Kim and H. J. Kim. Unsupervised moving object segmentation and recognition using clustering and a neural network. Proceedings of the 2002 International Joint Conference on neural networks (IJCNN '02), 2002, 2: 1240-1245.

[8] C. Carson, S. Belongie, H. Greenspan and J. Malik. Region-based image querying. Proceedings IEEE Workshop on Content-Based Access of Image and Video Libraries, 1997: 42 -49.

[9] A. B. Benitez. Semantic knowledge construction from annotated image collections. Image and Expo, 2002. 2: 205-208.

[10] M. Simard, S. S. Saatchi and G. DeGrandi. The use of decision tree and multiscale texture for classification of JERS-1 SAR data over tropical forest. IEEE Transactions on Geoscience and Remote Sensing, 2000, 38 (5): 2310–2321.

[11] H. Samet. The Quadtree and related hierarchical data structures. ACM computing surveys, 1984, 16 (2): 187-260.

[12] A. Guttman. R-trees: A dynamic index structure for spatial searching. ACM SIGMOD international conference on Management of data, 1984.

[13] N. Beckmann, H. Kriegel, R. Schneider, and B. Seeger. The R*-tree: An Efficient and Robust Access Method for Points and Rectangles, Proceedings of the ACM SIGMOD international conference on Management of data, 1990, 19 (2): 322-331.

[14] N. Katayama, S. Satoh. The SR-tree: An Index Structure for High-dimensional Nearest Neighbor Queries, Proceedings of the ACM SIGMOD international conference on Management of data, 1997, 26 (2): 369-380.

[15] J. T. Robinson. Physical Storage Structures: The K-D-B-Tree: A Search Structure for Large Multidimensional Dynamic Indexes. Proceedings of the ACM SIGMOD international conference on Management of data, 1981: 10-18.

[16] A. W. Fu, P. M. Chan, Y. Cheung, and Y. S. Moon. Dynamic VP-tree indexing for n-nearest neighbor search given pair-wise distances. VLDB Journal, 2000, 9(2): 154–173.

[17] W. Hsu, T. S. Chua, and H. K. Pung. Approximating content-based object-level image retrieval, Image Tools and Applications, 2000, 12 (1): 59-79.

[18] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos. The Bayesian image retrieval system, PicHunter: theory, implementation, and psychophysical experiments,, IEEE Transactions on Image Processing, 2000, 9 (1): 20–37.

[19] M. W. Bright, A. R. Hurson, and S. Pakzad. A Taxonomy and Current Issues in Multidatabase Systems. IEEE Computer, 1992, 25 (3): 50–60.

[20] M. R. Rezaee, P. M. J. van der Zwet, B. P. E. Lelieveldt, R. J. van der Geest, and J. H. C. Reiber. A Multiresolution Image Segmentation Technique Based on Pyramidal Segmentation and Fuzzy Clustering. IEEE Transactions on Image Processing, 2000, 9 (7): 1238-248.

[21] B. Heisele and W. Ritter. Segmentation of Range and Intensity Image Sequences by Clustering. 1999 International Conference on Information Intelligence and Systems, 1999: 223–225.

[22] D. Yu and A. Zhang. Clustertree: Integration of Cluster Representation and Nearest Neighbor Search for Image Databases. 2000 IEEE International Conference on Image and Expo, 2000, 3: 1713-1716.

[23] Y. Konig and N. Morgan. Supervised and unsupervised clustering of the speaker space for connectionist speech recognition. 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-93), 1993, 1: 545-548.

[24] Jia Wang, W. Yang and R. Acharya. Color clustering techniques for color-content-based image retrieval from image databases. IEEE International Conference on Image Computing and Systems, 1997: 442–449.

[25] J. Malik, D. A. Forsyth, M. M. Fleck, H. Greenspan, T. Leung, C. Carson, S. Belongie, and C. Bregler. Finding objects in image databases by grouping. International Conference on Image Processing, 1996, 2: 761-764.

[26] S. D. MacArthur, C. E. Brodley, and C. Shyu. Relevance feedback decision trees in content-based image retrieval. IEEE Workshop on Content-based Access of Image and Video Libraries, 2000: 68 –72.

[27] I. K. Park, etc. Perceptual grouping of 3D features in aerial image using decision tree classifier. 1999 International Conference on Image Processing (ICIP 99), 1: 31-35.

[28] A. Namasivayam. The use of fuzzy rules in classification of normal human brain tissues. The Third International Symposium on Uncertainty Modeling and Analysis and Annual Conference of the North American Fuzzy Information Processing Society (ISUMA - NAFIPS '95), 1995: 157 -162.