A Knowledge-Based Approach for Retrieving Images by Content*

Chih-Cheng Hsu, Wesley W. Chu
Computer Science Department
University of California, Los Angeles
Los Angeles, CA 90024
{ccheng,wwc}@cs.ucla.edu

Ricky K. Taira
Department of Radiology Sciences
University of California, Los Angeles
Los Angeles, CA 90095
rtaira@mail.rad.ucla.edu

August 1, 1996

Abstract

A knowledge-based approach is introduced for retrieving images by content. It supports the answering of conceptual image queries involving similar-to predicates, spatial semantic operators, and references to conceptual terms. Interested objects in the images are represented by contours segmented from images. Image content such as shapes and spatial relationships are derived from object contours according to domain-specific image knowledge.

A three-layered model is proposed for integrating image representations, extracted image features, and image semantics. With such a model, images can be retrieved based on the features and content specified in the queries.

The knowledge-based query processing is based on a *query relaxation* technique. The image features are classified by an automatic clustering algorithm and represented by Type Abstraction Hierarchies (TAHs) for knowledge-based query processing. Since the features selected for TAH generation are based on context and user profile, and the TAHs can be generated automatically by a clustering algorithm from the feature database, our proposed image retrieval approach is scalable and context-sensitive. The performance of the proposed knowledge-based query processing is also discussed.

*This work is supported in part by the National Science Foundation Scientific Database Initiative, Grant IRI9116849 and also in part by ARPA contract F30602-94-C-0207.

1 Introduction

Retrieving images by content is a key technology for image databases. Pixel matching methods employed for content-based retrieval are time-consuming and of limited practical use since little of the image object semantics is explicitly modeled. QBIC [18] uses global shape features such as area and circularity to retrieve similarly shaped objects. However, due to the limited precision of global shape features [15], such an approach has limited expressiveness for answering queries with conceptual terms and predicates. VIMS [1] retrieves similar images by relaxing feature values of the target image based on the standard deviation of the features. Independent of the target data values, the same amount of relaxation is applied on the target data values to represent the similarity of data. Such interpretation of similarity is not sensitive to the location of the target data values inside their value range. In an image data space, many features are based on multiple attributes. For example, *location* requires at least two attributes (i.e., positions on x-axis and y-axis). Using a standard deviation to interpret the variation of multi-attribute features lacks the consideration of correlation among different attributes.

In addition to the shape features of image object, spatial relationships between objects are also important. For example, Chang et al. [4] models the distribution of image objects using orthogonal spatial relationships. Chu et al. [7] models both the orthogonal and topological spatial relationships. To support image retrieval and ranking based on spatial relationship similarity, we need models that allow images with similar spatial relationships to be further compared and ranked.

Currently, images cannot be easily or effectively retrieved due to the lack of a comprehensive data model that captures the structured abstracts and knowledge needed for image retrieval. To remedy such shortcomings, we propose a Knowledge-based Spatial Image Model (KSIM) which supports queries with *semantic* and *similar-to* predicates. *Semantic* predicates contain semantic spatial relationship operators (e.g., `INSIDE`, `NEARBY`, `FAR_AWAY`, etc.) and/or conceptual terms (e.g., `large`, `small`, etc.). The *similar-to* predicates allow users to retrieve images that are closely correlated with a given image based on a prespecified set of features.

We use an instance-based knowledge discovery technique, MDISC [6], to cluster similar images based on the user-specified image features (e.g., shape descriptors and spatial relationships). The knowledge required for resolving the meaning of *similar-to* and semantic operators is called *image content interpretation knowledge*, and is represented based on the generated clustering knowledge. MDISC can acquire more comprehensive *image content interpretation knowledge* than that acquired by other multi-dimensional indexing techniques, such as K-D-B-tree (used in FIBSSR [17]) and R^* tree (used in QBIC [18]). This is because MDISC classifies images based on conceptual difference of the feature values, while K-D-B-tree and R^* tree cluster data based on minimizing the number of disk access per data retrieval. In addition, these clustering techniques do not consider the semantic difference of image features; thus no global conceptual view of the image clustering can be provided to represent conceptual predicates such as `LARGE` *tumor* and *tumor* `NEARBY` *an organ*.

This paper is organized as follows: Section 2 presents the Knowledge-Based Spatial Image Model (KSIM) which integrates the image representations, extracted image features, and knowledge representing image semantics and similarity. Section 3 discusses the methodology of extracting image object features, such as shape features and spatial relationships, from the object contours. Section 4 presents a methodology to extend existing query languages for including the proposed operators, and Section 5 describes the required intelligent interpretation and access. Section 6 presents our knowledge-based query processing technique, and Sections 7 and 8 present the performance results and our conclusions.

2 The Knowledge-Based Spatial Image Model (KSIM)

A three-layered image model is used to integrate the image representations and image features together with *image content interpretation knowledge*. The three layers are the Representation Layer (RL), the Semantic Layer (SL), and the Knowledge Layer (KL). Each layer consists of its own constructs, and these constructs are linked for cross-reference.

Raw images are stored in the RL where multiple representations of the same image objects may exist (e.g., X-ray images, magnetic resonance images, CT images, etc.). Image objects that can be queried are represented by contours in the RL. The contours can be segmented manually, semi-automatically (e.g., using techniques like snake [12] and flooding in [18]), or automatically [25, 24] depending on the contrast and separability of the image objects. Computing image features based on known object contours rather than based

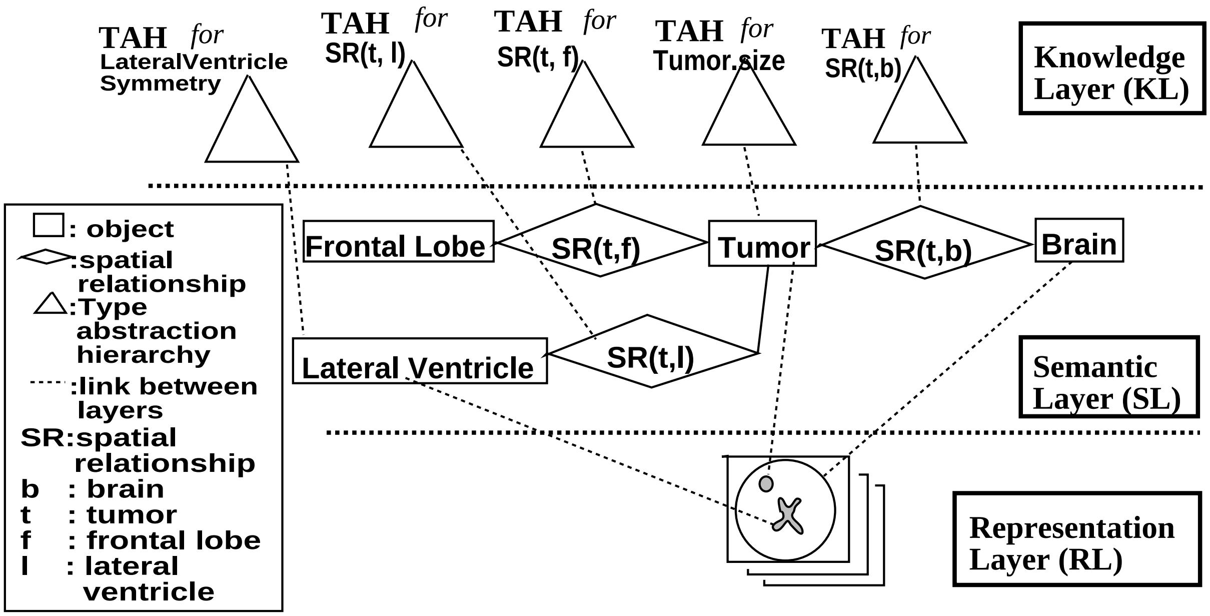


Figure 1: An example representing the brain tumors in KSIM. SR(t,b), SR(t,l), and SR(t,f) represent the spatial relationships between tumor and brain, tumor and lateral ventricle, and tumor and frontal lobe. The detailed TAH for lateral ventricle is shown in Figure 3, and the TAH for SR(t,l) is shown in Figure 6.

on raw images results in features of high certainty. Features of high certainty avoid the probabilistic interpretation of image features [21]. Contour segmentation routines [25, 12, 14, 24] are available to assist in identifying object contours from raw images.

Despite the enormous efforts toward automatic segmentation of medical images, success has been limited to only a few types of medical objects. These objects, in general, have high contrast with respect to their background (e.g., bones in projectional X-rays and computed tomography, and arteries with contrast agents in X-ray angiography), relatively simple shapes (breast outline in a mammogram), sizes that are not too small, and little or no overlap with other objects (e.g., central cross-sectional slice of lateral ventricle of the brain). In general, large medical image repositories (*e.g.*, radiological picture archiving and communication systems) contain diverse instances of complex image objects (anatomy and pathology), and thus automated segmentation of these objects are the bottleneck for the large-scale deployment of our technique. The emergence of more intelligent segmentation routines that use various physical models of the target objects (e.g., lungs and bronchial tree) [2, 20, 23] to assist in object delineation may result in a greater number of robust and automated medical image object identification programs.

In the SL, an object-oriented technique is used to model the image content extracted from the image representations in the RL. Image objects are modeled as *feature objects*. Spatial relationships among objects are represented by their *spatial relationship features* such as distance of centroids, ratio of overlapping area, etc. Features in the SL are computed from image object contours by the *shape model* and *spatial relationship model*. The shape model computes the required shape features, and the *spatial relationship model* computes the required spatial relationship features. Object-oriented inheritance hierarchies are used to organize similarly related objects.

In the SL, features are classified into *derived features*, *composite features*, and *conceptual features*. *Derived features* are features extracted from the corresponding contour(s) (e.g., area of an object contour) or derived from other *features* (e.g., the ratio of perimeter to area of a contour). A *composite feature* combines several features into a multi-attribute feature to reflect the specific content of an object. For example, the composite feature *location* of an image object consists of the *x_location* and *y_location* of the contour's centroid. A *conceptual feature* is a *composite* or *derived* feature with appended knowledge to represent the image semantics or similarity based on the feature.

The knowledge layer (KL) contains the logic for interpreting image semantics and image similarity based on the extracted image feature values. Type abstraction hierarchies (TAHs) [8, 5, 9], which represent general image concepts in the higher levels and specific concept in the lower levels, are used to represent the knowledge of the selected object features and spatial relationships. TAHs provide a way to represent the image semantics and similarity. Figure 1 illustrates the three-layered modeling and the linking among the representation of image objects (i.e., contours), semantic relationships among the objects, and knowledge required for representing brain tumors.

The features of contoured image objects in a database are extracted according to the *shape model* and *spatial relationship model* and stored as a *feature database*. These features are then classified by a conceptual clustering algorithm, MDISC [6], and the feature classification hierarchy is represented in TAHs which provide a multi-level knowledge representation of the image content based on analyzed features. Such TAHs are used to process queries with semantic operators (e.g., *"Find a* `large` *tumor* `NEARBY` *the lateral ventricle"*) and queries with *similar-to* operator (e.g., *"Find patients with similar brain tumors to pa-*

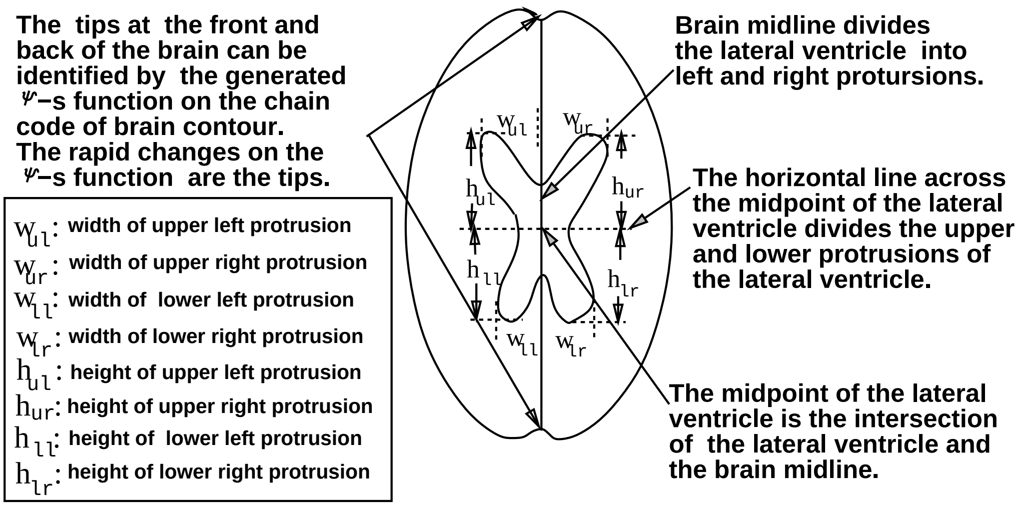


Figure 2: The shape model decomposes a lateral ventricle into four natural sub-structures for more precise shape description: upper left protrusion, upper right protrusion, lower left protrusion, and lower right protrusion.

tient with id `'P000-001'` *based on the tumor size and the location of the tumor* `NEARBY` *the lateral ventricle"*). The conceptual terms (e.g., `large` and `NEARBY`) can be translated to value ranges of relevant features via TAHs. For example, the value range representing `large`-sized tumor can be derived from the TAH for tumor size, and the value ranges representing `NEARBY` can be derived from the TAH that specifies the spatial relationship between tumor and lateral ventricle (i.e., SR(t,l)). For *similar-to* operator, based on the query context and user behavior, a set of relevant features representing the similarity of the target image is selected. The appropriate TAHs that represent these selected features can be used to derive the feature value ranges of the images that are most similar to the target image. These derived value ranges are used as the query constraints for retrieving the similar images. The methodology for extracting features and spatial relationships from object contours is presented in Section 3, and the methodology for generating the required knowledge is presented in Section 5.

3 Capturing Object Shape and Spatial Relationships

The *shape model* and *spatial relationship model* in the SL are used to extract image features from contours.

object feature	*conceptual terms*
tumor.size	*small, medium, large*
tumor.roundness	*circular, non_circular*
lateral_ventricle.left_to_right_symmetry	*symmetric* *upper_protrusion_pressed_to_the_right* *upper_protrusion_pressed_to_the_left* *lower_protrusion_pressed_to_the_right* *lower_protrusion_pressed_to_the_left*
...	...

Table 1: A *shape feature description table* for the brain

3.1 Modeling Shape

Shape of a contour can be described quantitatively using numeric shape descriptors such as *roundness*, *curveness*, *rectangularity*, *compactness*, *direction*, *elongatedness*, and *eccentricity* [22]. These descriptors are called *shape features* of the image objects. These shape descriptors provide a global description of object shape, but lack detailed variations [15]. We propose a two-staged approach to capture the shape content. In the first stage, complex contours are decomposed into context-dependent natural sub-structures based on the fundamental line and curve segments identified by the generated $\psi - s$ function from the chain code of the relevant object contours [16, 19]. For example, the lateral ventricle is decomposed into four protrusions based on the two tips of the brain contour found by the $\psi - s$ function from the brain contour as shown in Figure 2. In the second stage, these more elementary contour components are characterized by their shape features such as *area*, *height*, and *width*. Thus, we can express the shape and spatial relationships among these decomposed contours to reflect the specific shape content of the image object. This two-staged shape description allows more specific and detailed shape description using numerical shape descriptors rather than applying shape descriptors directly [18]. For example, in Figure 2 the height and width of the four components of a lateral ventricle are used to construct a multi-attribute shape feature to describe the left to right symmetry of the lateral ventricle as (upperLRWidthRatio (w_{ul}/w_{ur}), upperLRHeightRatio (h_{ul}/h_{ur}), lowerLRWidthRatio (w_{ll}/w_{lr}), lowerLRHeightRatio (h_{ll}/h_{lr})). Grouping features (e.g., length, width, height, area, *etc.*) from the decomposed components forms a *composite feature* that describes the detailed shape characteristics of the contour.

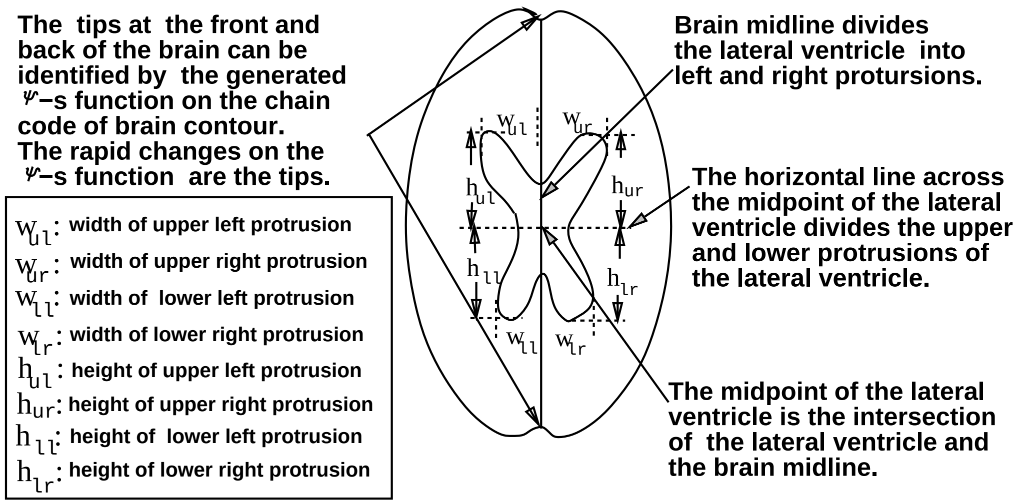


Figure 3: Multi-attribute Type Abstraction Hierarchy (generated by MDISC based on the decomposed four protrusions) representing the left to right symmetry of the lateral ventricles

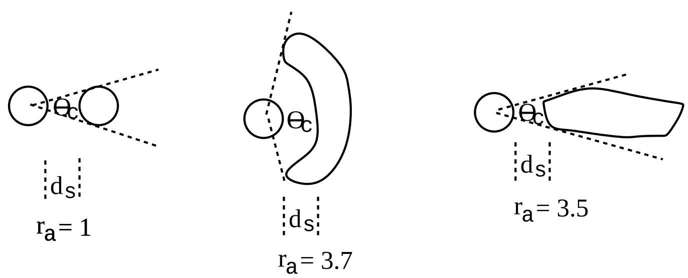


Figure 4: An example showing that using semantic operators (e.g., *non_overlapping*) and/or single measurement (e.g., the *shortest distance* (d_s)) is insufficient to capture the spatial relationship of two objects. We need additional features such as angle of coverage (θ_c) and ratio of area (r_a) to classify the illustrated spatial relationship.

Decomposition provides an effective quantitative shape description when the image objects have limited numbers of shape components. This description provides sufficient image content to retrieve similarly or specifically shaped image objects. Conceptual terms can be defined on a shape feature. The *shape feature description table* (Table 1) lists the available conceptual terms for the shape features in the system. Thus, users can ask queries with conceptual terms for a specific shape feature such as *"retrieving lateral ventricles whose upper protrusion are pressed to the right"* (see *Query 3* in Section 4).

3.2 Modeling Spatial Relationships

Modeling spatial relationships merely by simple semantic constructs such as *separated* and *connected* is insufficient to compare real-life spatial relationships (as illustrated in Figure 4).

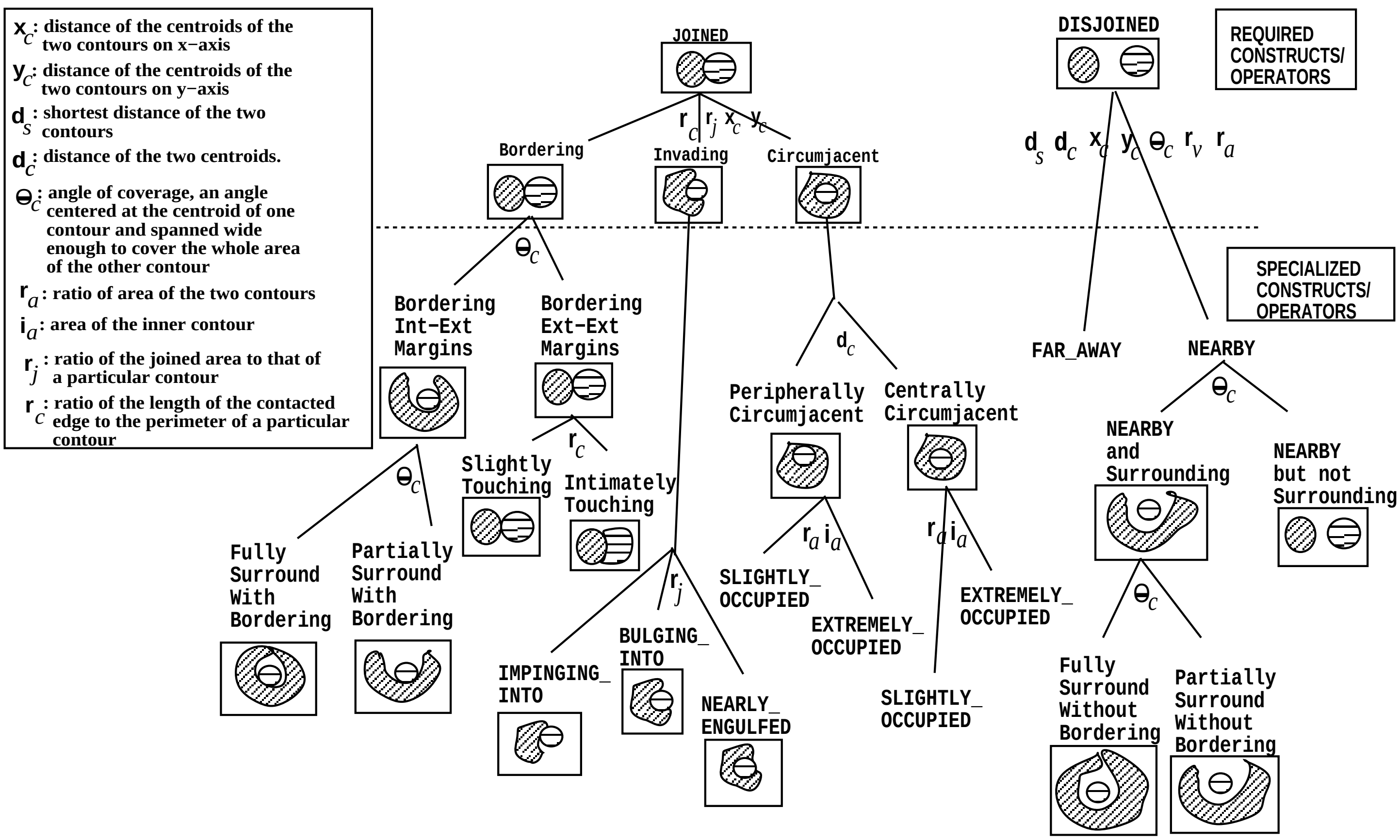


Figure 5: Semantic spatial relationship operators for different topological categories between two objects (with the representative icons shown). The parameters under a branch classify the sub-types under that category.

spatial relationship	*representative features*	*defined semantic terms*
SR(t,b)	$(x_c, y_c, r_a), (a_i)$	SLIGHTLY_OCCUPIED, EXTREMELY_OCCUPIED
SR(t,f)	(x_c, y_c, r_j)	SLIGHTLY_TOUCHED, INTIMATEDLY_TOUCHED
SR(t,l)	$(\theta_c, d_c, x_c, y_c)$	NEARBY, FAR_AWAY
...	...	...

Table 2: A *spatial relationship description table* for the brain tumor

Additional parameters are needed to more precisely describe the spatial relationships. A set of required spatial relationship features should be specified by domain experts, and the values of these spatial relationships are stored in the database. In Figure 5, useful parameters are illustrated with their importance in distinguishing the topoligical relationships between two objects. More important parameters for distinguishing the sub-types under a category are placed first in the list, and parameters appearing at higher branches may also be used in their decendant branches. In Figure 5, `BORDERING` means that only the surfaces of the two objects are joined (i.e., $r_c > 0, r_j = 0$); `INVADING` implies that their areas are joined (i.e., one of the object is deformed by the other, $0 < r_j < 100\%$); and `CIRCUMJACENT` implies that $r_j = 100\%$. The *required operators* are necessary for every spatial relationship.

In an image with a tumor and lateral ventricle, for example, the spatial relationship instance between the tumor and lateral ventricle is classified as an instance of the class SR(t,l). This spatial relationship requires θ_c, d_c, x_c, and y_c to represent it. These values are computed based on the object contours. The *spatial relationship description table* (as shown in Table 2) lists the representative parameters and available semantic terms for the spatial relationships in the system.

Figure 6 is an image classification hierarchy of images in the database which is generated by MDISC based on spatial relationship features of SR(t,l) where two operators `NEARBY` and `FAR_AWAY` are defined. With this spatial relationship modeling, a richer set of spatial relationship parameters not only enhances the quality of the (context-senstive) semantic spatial relationship operators, but also provides suitable parameters to be considered for resolving `SIMILAR_TO` operators in comparing spatial relationships.

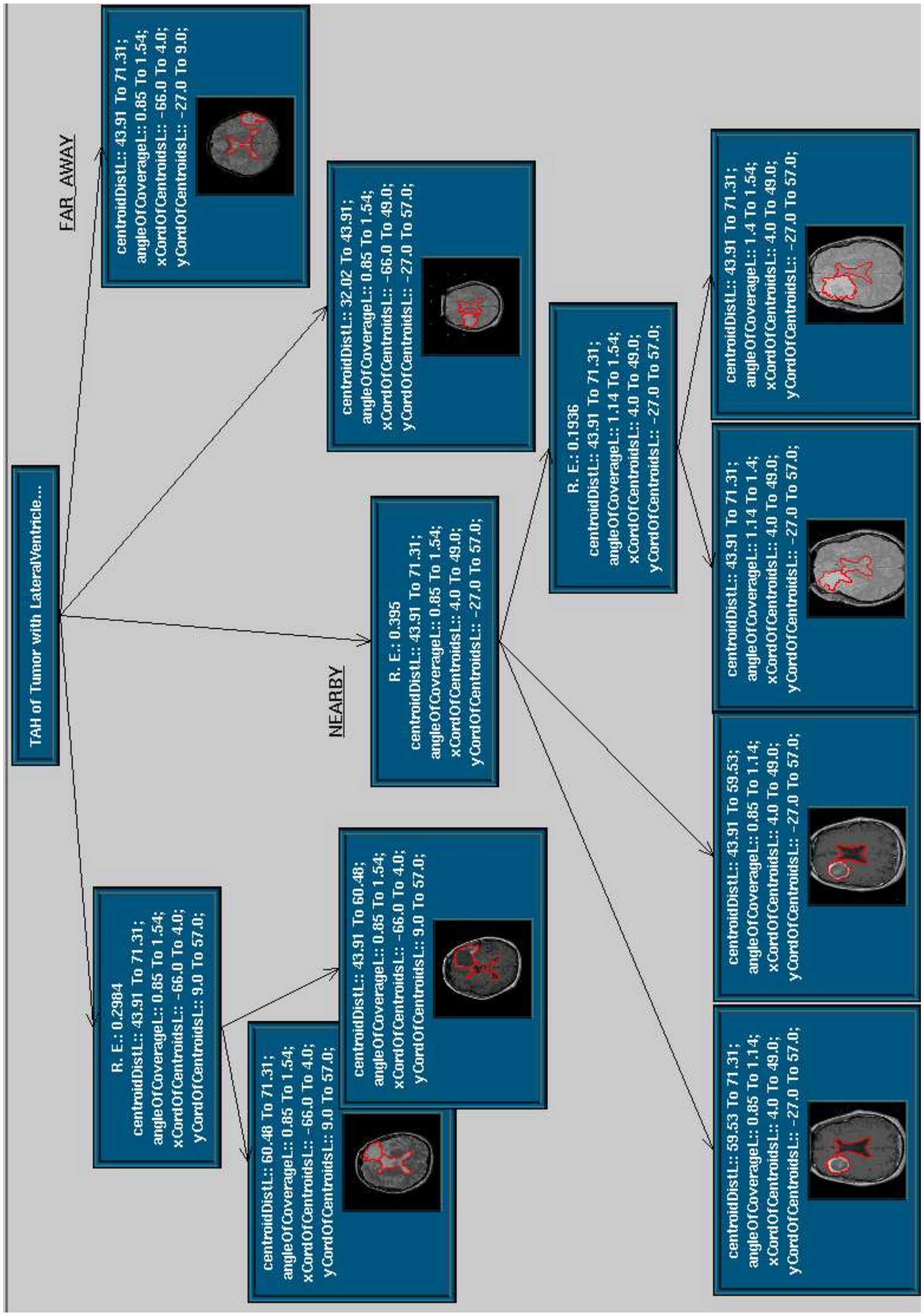


Figure 6: The MDISC-generated TAH for representing the spatial relationship between tumor and lateral ventricle. The TAH is generated based on d_c, θ_c, x_c, and y_c (denoted as centroidDist, angleOfCoverage, xCordOfCentroids, and yCordOfCentroids in the figure).

4 Extending Query Language with Knowledge-based Spatial Query Constructs

We shall now present the BNF specification for extending an object-oriented query language, such as OQL-93 [3], to include the proposed three types of predicates: (1) `SIMILAR_TO` *predicates*, (2) *semantic spatial relationship predicates*, and (3) *predicates with conceptual terms*. A similar extension for SQL was explored in CoBase [10, 9] for transportation and GIS applications.

The `SIMILAR_TO` operator is used to search for objects similar to a specified target object `BASED_ON` a set of features specified in the query. The syntax of the `SIMILAR_TO` predicate in BNF is:

```
similar_to_pred   ::= object SIMILAR_TO object (target_obj_condition)
                                BASED_ON obj_features  |
                      image SIMILAR_TO image (target_image_condition)
                                [ BASED_ON spatial_aspects]

spatial_aspects   ::= spatial_aspect [","  spatial_aspects]

spatial_aspect      ::= spatial_relationship_feature | obj_feature

spatial_relationship::= DISJOINED|NEARBY|FAR_AWAY|NEARBY_and_SURROUNDING|
                        NEARBY_butNot_SURROUNDING|
                        PARTIALLY_SURROUND_without_BORDERING|
                        FULLY_SURROUND_without_BORDERING|JOINED|BORDERING|
                        BORDERING_Int-Ext_MARGINS|BORDERING_Ext-Ext_MARGINS|
                        PARTIALLY_SURROUND_with_BORDERING|
                        FULLY_SURROUND_with_BORDERING|INTIMATE_TOUCHING|
                        INVADING|IMPINGING_INTO|BULGING_INTO|NEARLY_ENGULFED|
                        CIRCUMJACENT|PERIPHERALLY_CIRCUMJACENT|
                        CENTRALLY_CIRCUMJACENT|SLIGHT_OCCUPIED|EXTREMELY_OCCUPIED

target_obj_condition   ::= object_pathlist = literal

target_image_condition ::= image_pathlist = literal | image SELECTED_ON_THE_SCREEN
```

The `object`, `obj_feature` and `spatial_relationship_feature` correspond to the semantic object, object features, and spatial relationship features in the SL. The `image` refers to an image from which a collection of image objects are extracted for querying and comparison. The `BASED_ON` subclause specifies the shape features (i.e., `obj_feature`) and/or specific spatial relationships between objects (i.e., `object spatial_relationship object`) that represent the intended similarity of the query. If no BASED_ON subclause is specified, the knowledge in the KL determines the features that represent the similarity based on the query context and user type. target_object_condition and target_image_condition specify the path condition (e.g., image.patient.ID) to select a distinct target object or image to be compared with where literal is a constant. SELECTED_ON_THE_SCREEN is a special function used to specify an image on the screen as the target image for matching.

The syntax for the *semantic spatial relationship predicates* is:

```
sr_pred = object spatial_relationship object
```

To avoid ambiguity in specifying the operators, a pull-down menu is available that display the available specialized operators as in the *spatial relationshp description table* (Table 2) for the user to select a suitable operator to be used in the query.

The syntax for the predicate expressed with conceptual term(s) is:

```
obj_feature  IS conceptual_term
```

Likewise, a pull-down menu is also used to display the available conceptual terms for the specified obj_feature as in the *shape feature description table* (Table 1). The conceptual_term is interpreted by the knowledge residing in the KL [5, 9].

Example Queries

Query 1: "Find patients with similar brain tumors to the patient with id 'P000-001' *based on the tumor size and tumor location* NEARBY *lateral ventricle."*

```
select   patientWithImage( patient: i1.patient, image: i1.image)
from     Images i1, it
where    i1 SIMILAR_TO it ( it.patient.id = 'P000-001' )
           BASED_ON (it.tumor.size,
                     it.[tumor,lateral_ventricle].(x_c, y_c, θ_c, d_c))
```

patientWithImage is a constructed type for displaying query results [3].

Query 2: "Find large *tumor* NEARBY *the lateral ventricle."*

```
select patientWithImage( patient: t.patient, image: t.image)
from   Tumors t, Lateral_Ventricles l
where  t NEARBY l and
       t.size IS 'large'
```

Query 3: "Find the lateral ventricle whose upper protrusion is pressed to the right."

```
select patientWithImage( patient: l.patient, image: l.image)
from   Lateral_Ventricles l
where  l.left_to_right_symmetry
          IS 'upper_protrusion_pressed_to_the_right'
```

The knowledge representing `upper_protrusions_pressed_to_the_right` is provided in Figure 3.

A brain surgeon wishes to retrieve images of patients in the database with similar spatial characteristics as the presented MR image. The textually expressed query is shown in *Query 4*, and a graphical expression of the same query is illustrated in Figure 11 in Section 6.

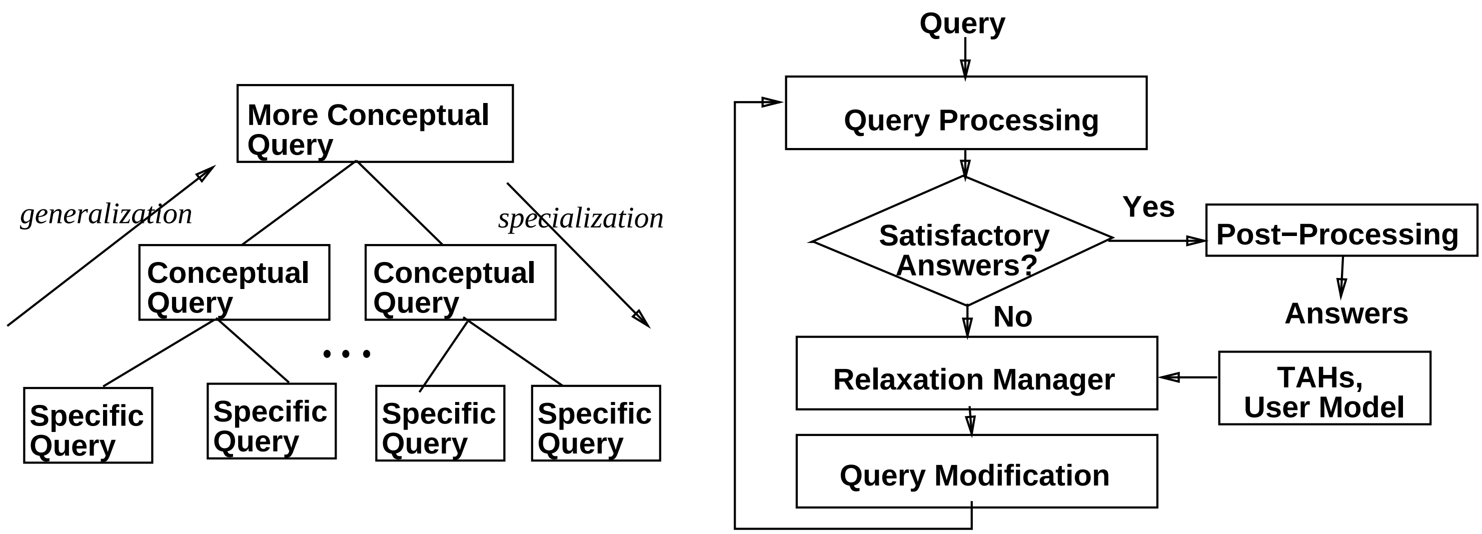
Query 4: "Find images in the database that have similar spatial characteristics as the given image on the screen."

```
select   patientWithImage( patient: p1, image: p1.image)
from     Patients p1, Patients pt
where    p1.image SIMILAR_TO pt.image (pt.image SELECTED_ON_THE_SCREEN)
```

The intended features and spatial relationships of *Query 4* are derived by the knowledge layer based on the image content in `PT.image` and the user type (i.e., brain surgeon).

5 Intelligent Interpretation and Access

The criteria of our image feature clustering algorithm is to minimize the averaged pair-wise euclidean distance of image feature values in a cluster. Such a measure, known as the *relaxation error* [6], considers both the *frequency* of the value occurrence and the difference between *values*. Based on minimizing the summed *relaxation error* of all the new partitioned clusters in each iteration, the clustering algorithm, MDISC, recursively partitions the data set to generate a multi-attribute feature type abstraction hierarchy (MTAH). As both the feature value distribution and the correlation among different attributes of a feature are considered, our clustering algorithm provides better image feature classification than those using standard deviation to represent image similarity [1].



(a) Generalization and specialization via TAH **(b) The flow diagram of query processing with relaxation**

Figure 7: Knowledge-based query relaxation

5.1 Query Interpretation via TAH

The image classification hierarchies are represented in type abstraction hierarchies [8, 5, 9] for processing *similar-to* and semantic predicates. The concept in the TAH nodes is represented as the value ranges of the features (see Figure 3 and Figure 6). These value ranges can be used to retrieve similar images. As shown in Figure 7(a), higher nodes in the TAH represent more generalized concepts (i.e., wider range of feature values) than that of the lower nodes (i.e., narrower range of the feature values). The TAH nodes can be labeled with conceptual terms (e.g., `large`, `small`, `upper_protrusion_pressed_to_the_right`) to represent the specific knowledge. These available conceptual terms are listed in Table 1 to provide a pull-down menu for assisting users during query specification.

The knowledge of the semantic spatial relationship operators can also be represented by the TAH. Based on the topological relationships of two objects [13], useful semantic operators are shown in Figure 5. MDISC is used to classify image features for defining these semantic spatial relationship operators based on the values of the representative spatial relationship features. The resultant TAH nodes can be labeled with an appropriate subset of the detailed operators (e.g., `NEARBY, FAR_AWAY`) to represent the value ranges representing the semantic spatial relationship operators. These value ranges are used as the query constraint to retrieve images satisfying the conceptual predicates.

To solve a *similar-to* query whose intended similarity includes the features or spatial relationship classified by a TAH, the lower TAH nodes are attached with more specific value ranges. In solving the *similar-to* query, we shall first locate the TAH node that has a value range closest to that of the target image based on the selected features. By traversing up (i.e., generalizing) and down (i.e., specializing) the selected TAH, the feature value range in the finalized TAH node is used to modify the query constraints for retrieving similar images from the database, as shown in Figure 7(b). The TAH traversal is controlled either by user input or by relaxation policy provided in the user model.

There is a *TAH directory* in the system that stores such information as object names, sets of features, spatial relationships, user type, explanation about the emphasis or purpose of the TAH, etc. Based on this information, the system (or user) selects and retrieves the appropriate TAHs for processing the query. If the retrieved TAH does not match user's specification, it can be edited by the user to meet his/her application.

The time complexity to generate a multi-attribute hierarchy by MDISC is $O(m(n(log(n))))$, where m is the number of attributes, and n is the number of distinct instances used in generating the TAH [6]. Our experiment reveals that to generate a MTAH with about one hundred images based on four features takes a fraction of a second's processing time on a Sun Sparc 10 workstation.

5.2 User Model

In our knowledge-based query processing, user behavior is characterized by his/her concerns (including image objects, set of features, and spatial relationships), object matching policy, and the policies for relaxing query conditions when no satisfactory answer is found. These behaviors can be represented by a *user model* to customize the query processing. Different types of users can be represented by different user profiles in the model. Objects in the user profile are divided into *mandatorily matched objects* and *optional matched objects*. *Mandatorily matched objects* of a user profile must be matched with the query context for the user profile to interpret the query. *Optionally matched objects* provide guidance for additional matched features to enhance the query constraints. Such an option permits a partial matching of the user model and increases the matching occurrences. The relaxation policy describes how to relax the selected TAHs when no satisfactory answers are found,

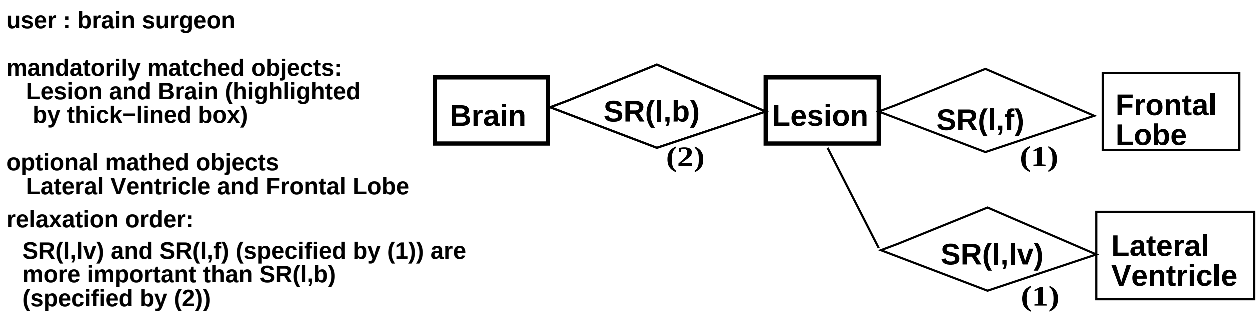


Figure 8: A user profile for brain surgeons

where each MTAH (such as SR(t,l) and SR(t,b)) represents different knowledge about the image objects. The relaxation policy specifies the *relaxation order* (e.g., which MTAH should be relaxed first), *relaxation level*, *non-relaxable objects*, etc. For more discussion on relaxation operators, interested readers should see reference [9].

In an MR brain image with tumor(s), for example, a brain surgeon's concerns regarding the brain tumors are their locations and the spatial relationships with other objects in the brain, as shown in Figure 8. The information in this user profile can be used for processing queries such as *"retrieve similar images as the brain tumor shown on the screen."* Different types of users (e.g., radiologists, surgeons, and clinicians) may have a different emphasis. Thus, different user profiles can be represented in the user model for the same set of images.

6 Knowledge-Based Query Processing

6.1 Query Processing

Query processing can be divided into three phases, as shown in Figure 9: the *query analysis and feature selection phase*, the *knowledge-based content matching phase*, and the *query relaxation phase*. In the *query analysis and feature selection phase*, based on the target image, query context, and user type, the system analyzes and selects the relevant features and spatial relationships for processing the query. For similar-to queries (i.e., path 1 in Figure 9 is selected), the features and spatial relationships specified in the BASED_ON subclause are the features representing the intended image similarity. If no BASED_ON subclause is specified, the user type and objects contained in the target image are used to select the features and spatial relationships representing the intended image similarity according to the matched user profile. After the intended features are selected, the shape

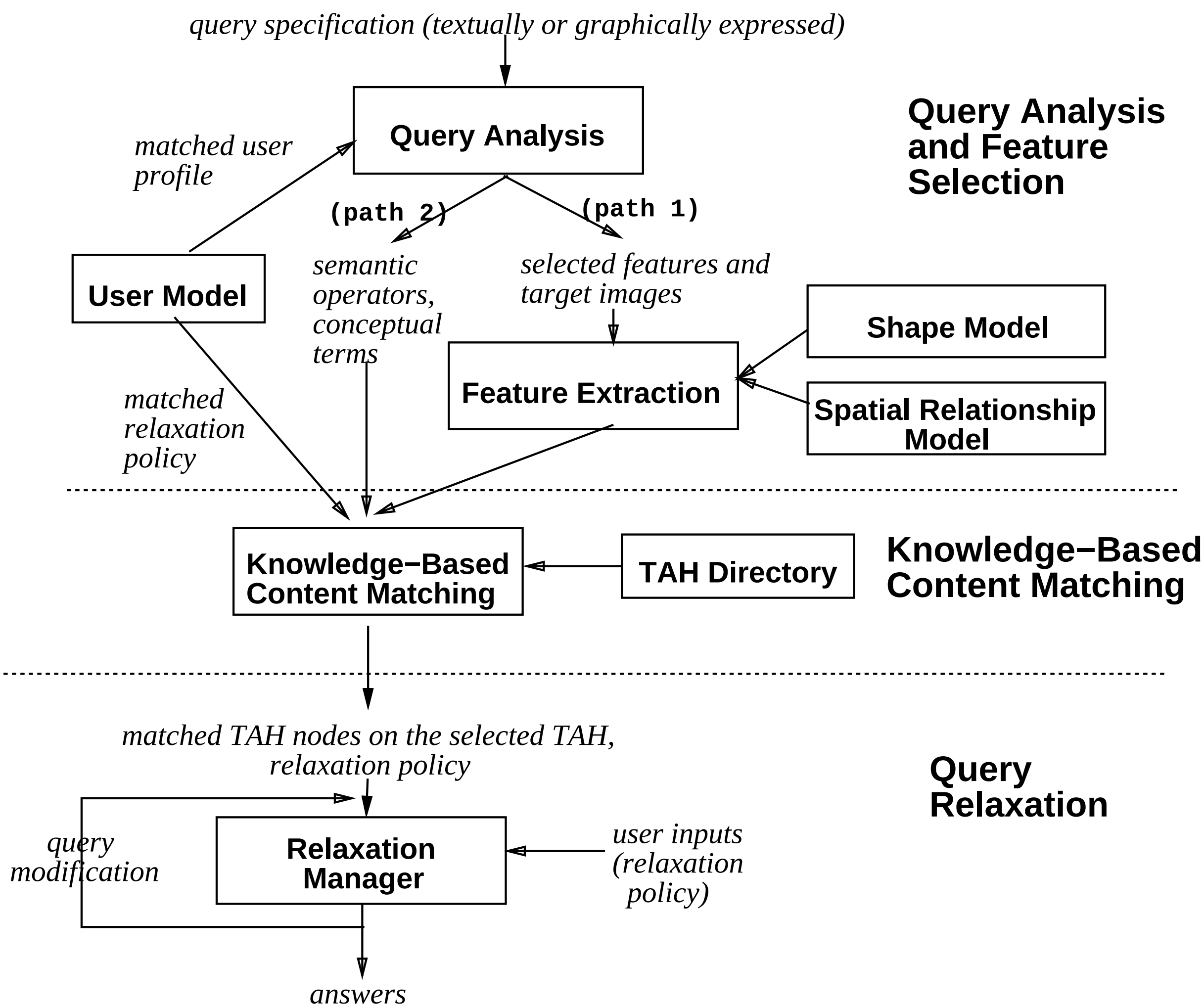


Figure 9: The flow diagram of knowledge-based query processing

and spatial relationship models extract their values from the object contours in the target image. For semantic queries (i.e., path 2 in Figure 9 is selected), the semantic spatial relationship predicates and conceptual terms in the query provide the selected features and spatial relationships.

In the *knowledge-based content matching phase*, the spatial relationship operators and conceptual terms are used to select the matched TAH(s) and TAH node(s) for processing the semantic queries. For *similar-to* queries, the selected features, spatial relationships, and user types are used to match TAH(s). The matched TAHs are traversed to locate the node with a value range closest to that of the target image. The set of images contained in the TAH nodes that has the closest matched value ranges represents the set of images similar to the target image.

In the *query relaxation phase*, the query is processed by traversing up and down the TAH(s) starting from the matched TAH nodes based on the relaxation policy provided in the matched user profile and user input. In every relaxation iteration, the query constraints are modified by the value ranges specified in the selected TAH nodes to retrieve the similar images. This relaxation process repeats until it reaches user satisfaction (e.g., number of similar images, relaxation error, etc. [5]). The returned images can be ranked based on the selected features. For the queries with semantic operators and/or conceptual terms, the value ranges in the finalized TAH nodes (i.e., the TAH nodes whose labels best match the semantic operators and/or conceptual terms) are used as the query constraints to retrieve the intended images. Since TAHs are user- and context-sensitive, the user can select the appropriate TAHs for his/her applications.

Figure 10 illustrates the query processing for a query with a *similar-to* operator where the target image is shown in the *target image canvas* of Figure 11. No `BASED_ON` subclause is provided in this example query, and the user model in Figure 8 is matched. The system allows user input to control the relaxation process which may overwrite the relaxation policy provided by the selected user model. According to the relaxation control specified in the user model, SR(t,l) is the first candidate TAH to be relaxed. Based on the TAH of SR(t,l) in Figure 6, the resulting value ranges for retrieving similar images are:

$(43.91 \leq SR(t,l).d_c \leq 71.31)$, $(0.85 \leq SR(t,l).\theta_c \leq 1.54)$,

$(4.0 \leq SR(t,l).x_c \leq 49)$, $(-27 \leq SR(t,l).y_c \leq 57)$

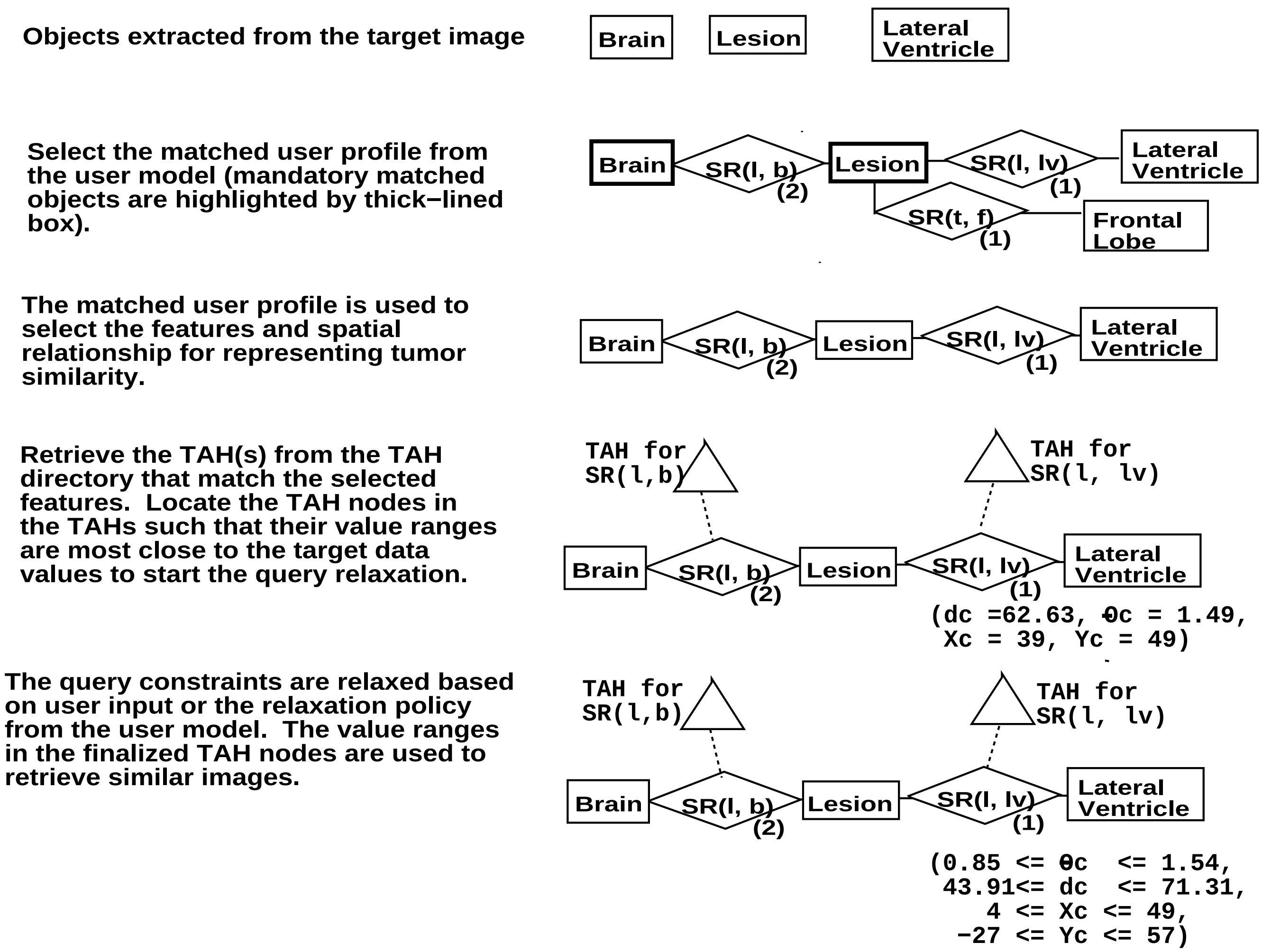


Figure 10: The query processing of Query 4 (the TAH of *SR(t.l)* is shown in Figure 6)

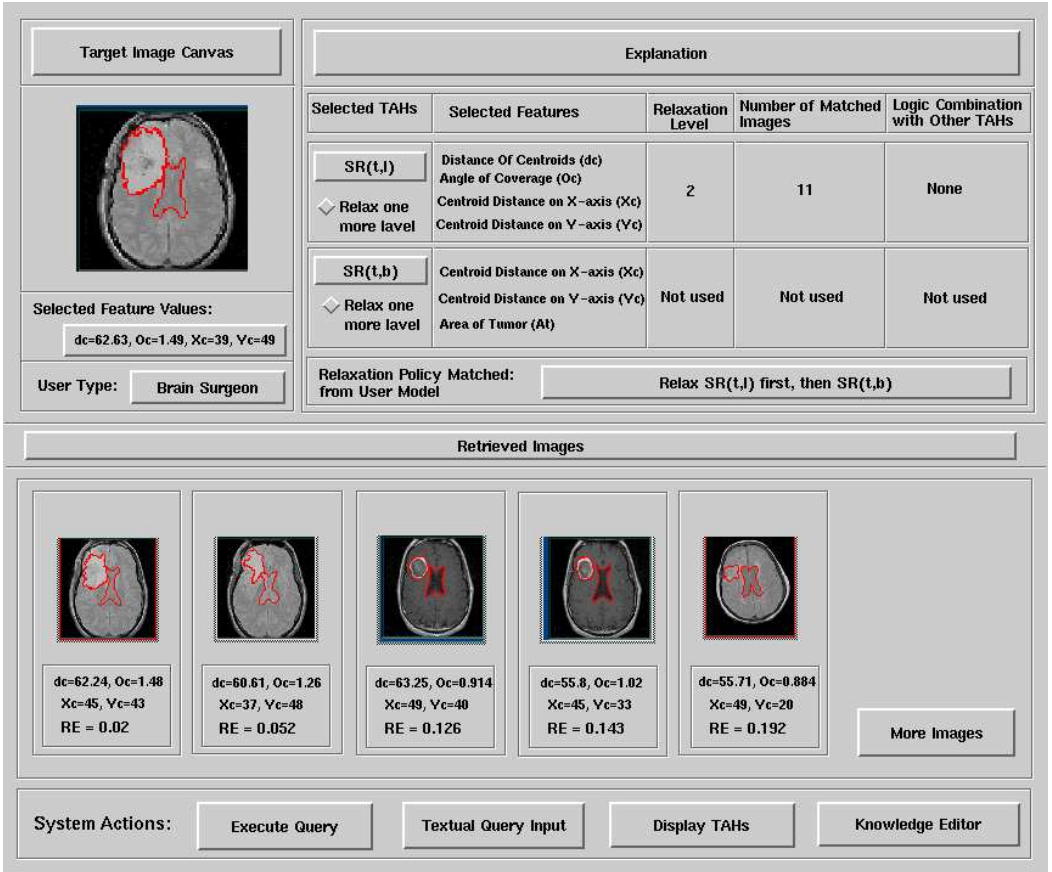


Figure 11: The graphical user interface (GUI) of the knowledge-based query answering

These value ranges correspond to the value range of the TAH node two levels higher from the matched leaf node.

The retrieved images are shown and ranked on the GUI with the *relaxation error* attached to each retrieved image. There is an *explanation* window which displays the selected features and spatial relationships used for the matching, the relaxation level, and the number of instances matched on the TAH node. During the relaxation process, if the relaxation of a TAH reaches a certain *relaxation error threshold* provided by the user model, then the system selects the next TAH for relaxation according to the relaxation policy. Users can also selectively combine the TAHs with logical operations (e.g., AND, OR, etc.) to retrieve the (desired) images.

7 Performance of the Knowledge-Based Query Processing

The TAH generation is based on the set of features used to classify objects in the images. For example, size and location are used in classifying images of brain tumors. The instances covered by the selected TAH node are candidates for matching the target image. Thus the set of features used for classifying affect the *precision of the retrieval* (i.e., retrieved relevant answers/all relevant answers). Using irrelevant features in classification will reduce the *precision of the retrieval.* For query with a `SIMILAR_TO` operator, the set of features used to compare the similarity affects the precision value. The weights assigned to the features reflect their relative importance in computing the similarity measure for ranking the retrieved images.

As the relationship among the objects in the image becomes more complex, more features are needed to specify the target images. For example, in specifying the characteristics of an object in an image, in addition to size, we can also include the shape and position of the object. In specifying the spatial relationship between two objects, in addition to their relative location and angle of coverage, the ratio of joining area or volume, and longest or shortest distance of the two objects can also be used in specifying additional characteristics of the target image. Therefore, using more precise specifications increases *precision of the retrieval.*

The *recall of retrieval* (retrieval relevant answers/all retrieval answers) depends on the relaxation error of the TAH node(s) of the referenced TAH(s) (i.e., the larger the relaxation error of a node, the lower recall value the TAH node yields) as well as the importance of the features in characterizing objects in the image. To increase the recall value, the range of the TAH nodes should be small (small relaxation error) and the selected TAH(s) for query processing should contain important attributes for characterizing the objects and their interrelationship in the image. Since TAHs can be customized based on user type and context, the user can select the set of features for generating the TAH(s) for processing a specific query and control the performance of the retrieval based on the complexity of objects in the image and the available features of the objects for classification.

We have collected image and computed features for brain tumor examples as described

		TAH(size)	TAH(size, location, angle_of_coverage)
Precision	without ranking	32.92%	73.33%
	with ranking	33.75%	82.96%
Recall	without ranking	27.43%	52.52%
	with ranking	28.13%	59.41%

Table 3: Performance of the knowledge-based query processing (in terms of precision and recall) for *Query 4* based on the two different TAHs

in query 4 in our prototype system. The images database consists of 65 magnetic resonance (MR) images (256 x 256 x 8 bits) containing brain tumors. Using the DISC algorithm, the images are classified into two TAHs: one based on tumor size and the other based on size, location, and the angle of coverage relative to the lateral ventricle. The *relevant answers* for each target instance are determined by exhaustively ranking all the images in the database by the similarity measurement based on the features selected by the domain expert (e.g., radiologists). Using the best-10 retrieving strategy (i.e., the generalization steps continue until the TAH node covers at least 10 instances) and taking each of the 65 images in the database as the target image, the average precision and recall values are shown in Table 3. This illustrates that the number of features used to specify the target image as well the ranking plays an important role in the performance of the retrieval.

The query response time includes the time for parsing, feature computation (this is needed only in the case when the features of the target image are not pre-computed), query processing, image retrieval, and image display. Our testbed uses the GemStone object-oriented database and VisualWorks as the application development tools running on a SPARC 10 SUN Workstation. The query response time for *Query 4* is as follows: parsing takes less than 1 second, feature computation takes around 12 seconds (for extracting features of the target image shown on the screen), knowledge-based query processing (i.e., selecting TAH nodes to match with features) takes about 1 to 2 seconds, image display takes about 3 to 5 seconds (depending on the number of returned images). Each relaxation processing (i.e., generalize and specialize TAH nodes to obtain sufficient number of images) takes about 0.5 seconds. Thus the time of the knowledge-based query processing is about 2 to 3 seconds which is relatively small compared to the time for feature extraction and image display.

8 Conclusions

In this paper, we present a knowledge-based approach for retrieving images by image features and content. The model supports semantic operators (e.g., `JOINED, NEARBY, FAR_AWAY`), similar-to operators, and references to conceptual terms (e.g., `LARGE, SMALL`) in the image queries.

The proposed KSIM model consists of three layers: the Representation Layer, the Semantic Layer, and the Knowledge Layer. These layers integrate the image representation (i.e., image contours) together with the knowledge required to capture image content and interpret the captured content to provide domain- and user-specific query.

Our model considers shape structure and shape features as well as spatial relationship features. These features can be automatically or semi-automatically extracted from the image contours and stored in a feature database. Based on the specified features and spatial relationships, the knowledge of image semantics and image similarity can be automatically generated by our conceptual clustering algorithm using the extracted features in the database. The knowledge is represented in a special knowledge structure, Type Abstraction Hierarchy (TAH), which is used in the query processing through a generalization/specialization process on the TAHs. The value ranges of the finalized TAH node are used to modify the query conditions for retrieving images. A user model is introduced to allow users to customize their requirement of query answering. The system also presents the quality of the answers measured in *relaxation error* to the user. Since the feature computation and knowledge acquisition are automated, our proposed technique is scalable.

A prototype image database system, KMeD [11], based on the proposed model has been implemented at UCLA using the GemStone/VisualWorks platform. Our preliminary result indicates that such a knowledge-based technique is a feasible and effective approach to retrieve images by features and content.

9 Acknowledgement

The authors would like to thank John David N. Dionisio for implementation of the graphical user interface of the query language, Christine Chih for her assistance in image segmentation, Kuorong Chiang and Timothy Plattner for developing the programs for generating

TAHs for images, and Prof. Alfonso Cardenas for his stimulating discussions during the course of writing this paper.

References

[1] J. R. Bach, S. Paul, and R. Jain. A visual information management system for the interactive retrieval of faces. *IEEE Transaction on Knowledge and Data Engineering*, Oct 1993.

[2] M. S. Brown, R. W. Gill, H. E. Talhami, L. Wilson, and D. B. Doust. Model-based assessment of lung structures: inferencing and control system. In *Proc. of the SPIE*, volume 2433, pages 167–178, 1995.

[3] R. G. G. Cattell, editor. *The Object Database Standard: ODMG - 93 (Release 1.2)*. Morgan Kaufmann, 1996.

[4] S. K. Chang, C. W. Yan, and D. C. Dimitroff. An intelligent image database system. *IEEE Transaction on Software Engineering*, May 1988.

[5] W. W. Chu and Q. Chen. A structured approach for cooperative query answering. *IEEE Transactions on Knowledge and Data Engineering*, 6(5), Oct 1994.

[6] W. W. Chu and K. Chiang. Abstraction of high level concepts from numerical values in databases. In *Proceedings of the AAAI Workshop on Knowledge Discovery in Databases*, Jul 1994.

[7] W. W. Chu, T. Ieong, and R. Taira. A semantic modeling approach for image retrieval by content. *The VLDB Journal*, 3(4):445–478, Oct. 1994.

[8] W. W. Chu, M. Merzbacher, and L. Berkovich. The design and implementation of Cobase. In *SIGMOD 1993*, pages 517–522, May 1993.

[9] W. W. Chu, H. Yang, K. Chiang, et al. A scalable and extensible cooperative information system. *Journal of Intelligent Information Systems*, 6(6), 1996.

[10] W. W. Chu, H. Yang, K. Chiang, B. Ribeiro, and Gladys Chow. CoGIS: A cooperative geographical information system. In *Proceedings of SPIE Conference on Knowledge-Based Artificial Intelligence Systems in Aerospace and Industry*, Orlando, FL., Apr 1994.

[11] Wesley W. Chu, Alfonso F. Cardenas, and Ricky K. Taira. A knowledge-based multimedia medical distributed database system — KMeD. *Information Systems*, 20(2):75–96, 1995.

[12] D. Daneels, D. van Campenhout, et al. Interactive outlining: An improved approach using active contours. In *Image and Video Storage and Retrieval, SPIE*, 1993.

[13] M. J. Egenhofer. Reasoning about binary topological relations. In *Advances in Spatial Databases, SSD*, 1991.

[14] I. Kapouleas. Segmentation and feature extraction for magnetic resonance brain image analysis. In *10th International Conference on Pattern Recognition*, pages 583–590, 1990.

[15] M. D. Levine. *Vision in Man and Machine*. McGraw Hill Publishing Co., Inc., 1985.

[16] W. N. Martin and J. K. Aggarwal. Computer analysis of dynamic scenes containing currilinear figures. *Pattern Recognition*, 11:169–178, 1979.

[17] R. Mehrotra and J. Gary. Similar-shape retrieval in shape data management. *IEEE Computer*, Sep 1995.

[18] Wayne Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC project: Querying images by content using color, texture, and shape. In *Storage and Retrieval for Images and Video Databases, SPIE*, 1993.

[19] W. A. Perkins. A model-based vision system for industrial parts. *IEEE Transactions on Computers*, C-27(2), Feb 1978.

[20] J. Qian, T. Mitsa, G. Sudaramoorthy, and E. A. Hoffman. 3D bronchial tree model and fractal analysis as tools for performance evaluation of different CT acquisition/reconstruction schemes. In *Proc. of the SPIE*, volume 2433, pages 113–120, 1995.

[21] F. Rabitti and R. Savino. An information retrieval approach for image databases. In *Proceedings of 18th VLDB Conference*, 1992.

[22] M. Sonka, V. Hlavac, and R. Boyle. *Image, Processing, Analysis and Machine Vision*, volume 1. Chapman & Hall Computing, 1993.

[23] M. Sonka, W. Park, and E. A. Hoffman. Validation of an enhanced knowledge-based method for segmentation and quantitative analysis of intrathoracic airway trees from three-dimensional CT images. In *Proc. of the SPIE*, volume 2433, pages 158–166, 1995.

[24] R. Wasserman, R. Acharya, et al. Multimodality tumor delineation via fuzzy fusion and deformable modelling. In *Proceedings of the SPIE, Medical Image: Image Processing*, 1995.

[25] Andrew J. Worth, Steve Lehar, and David N. Kennedy. A recurrent cooperative/competitive field for segmentation of magnetic resonance brain images. *IEEE Transactions on Knowledge and Data Engineering*, 4(2):156–161, Apr 1992.