A Multimedia Database System for Thermal Ablation Therapy of Brain Tumors

John David N. Dionisio† Alfonso F. Cárdenas† Robert B. Lufkin‡
Keith Black‡ Ricky K. Taira‡ Wesley W. Chu†

†Computer Science Department
‡Department of Radiological Sciences
University of California, Los Angeles

August 8, 1995

Abstract

A prototype multimedia medical database is described for supporting thermal ablation therapy of brain tumors. The database is a component of the KMeD project at UCLA, and links images to patient data in a way that permits the user to view and query medical information using alphanumeric, temporal, and feature-based predicates. The database is linked to various visualization programs that permit the user to view and/or annotate the query results in various ways. The ability to manage and access multimedia information on the progress and reaction of tumors to various therapy protocols is a major requirement for the success of thermal ablation therapy. A similar approach is being undertaken for supporting different therapies of other types of tumors, thus showing the generality of our approach.

1 Introduction

Magnetic resonance (MR) guided thermal ablative therapy of brain tumors is a new technique that combines the diagnostic power and soft tissue contrast resolution of MRI with aggressive interventional methods to produce a new treatment model that has the potential to become a major defining therapy of the 21st century. Alternative therapies for brain tumors include conventional craniotomy and radiation therapy, both of which have disadvantages in terms of cost and invasiveness when compared to this newer technique.

In order to fully evaluate the potential of the new therapy and appropriately apply it to patients, it is essential to have access to the prior patient treatment data in a convenient format.

The database system described in this paper maintains the patient records, models, and images that are relevant to the thermal ablation therapy application domain. It represents a typical application of the technology that has been developed in the Knowledge-Based Multimedia Medical Distributed Database (KMeD) project at the University of California, Los Angeles [CCT95]. Using the modeling constructs provided by KMeD, our database can store patient records and images, associate them with each other, and perform queries on this data based on features such as tumor volume or histology.

Section 2 describes thermal ablation therapy in greater detail. Section 2.1 describes typical queries that this application would like to answer. Section 3 describes the database's logical model. Section 4 outlines the user interface and illustrates the interaction with a walkthrough of a sample query. Section 5 describes and illustrates the display and visualization of query results. We conclude in Section 6 with an overview of further research that is currently on-going within the KMeD laboratory.

2 Description of the Application

Thermal ablation therapy is the use of focal heating for the treatment of tumors. Techniques for thermal ablation of brain tumors were pioneered at UCLA [Arr60, SMH60, Zer65, ZK72, Aea91, CSL92, ADB+93, ZWL+93, ALD+95]. The procedure is particularly important in the treatment of brain tumors, where invasive surgery is either impossible or poses the risk of severe brain damage. Using specially designed interventional magnetic resonance instruments, a radiofrequency (RF) electrode is directed into the tumor with MR guidance. Instead of the usual surgical craniotomy exposure, a 2mm twist drill hole is used for access in the skull of the patient, who remains awake during the procedure.

Local anesthesia is used for skin, scalp, periosteum, and dura — the brain itself does not have pain receptors. With the assistance of MR visualization, the tumor is destroyed as the local temperature of the tissue is heated above 60° C, necessary to denature proteins and cause coagulation necrosis. Patient demographics, treatment information, and imaging-derived information about the lesion before, during, and after treatment are stored in the database for manipulation and retrieval.

2.1 Sample Queries

The development of the database described in this paper was driven by the information system requirements demanded to properly support research into and treatment of brain tumors. Implementation work to date has resulted in our ability to answer the following types of queries:

- What is the volume of the enhancing portion of the treated area?
- Which patients have tumors that grew by more than 70% after tumor ablation treatment?
- Retrieve other tumor cases whose histology classification, patient age, and location in the brain are similar to the one currently displayed.
- Graph the growth of this patient's tumor over time.

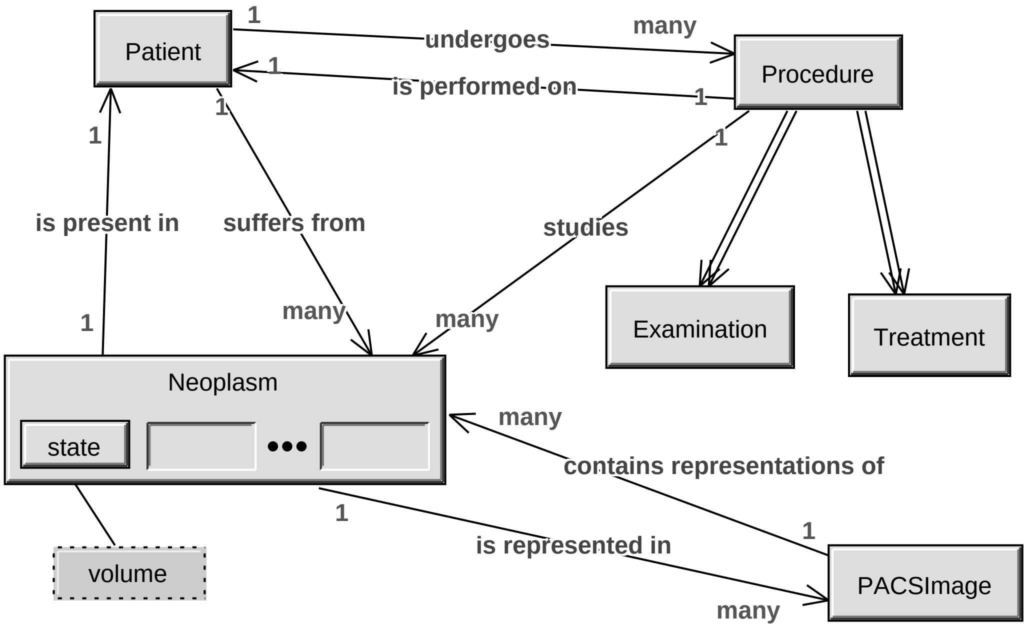

Figure 1: Logical model for the thermal ablation database.

Future work on the thermal ablation database will be driven by the need to correlate simulation data, images, and actual measurements. While fulfilling some of these requirements need new technologies that our group is still in the process of investigating (similar shape matching, for example), other functionalities are hindered only by the current unavailability of data, such as those generated by simulation and mathematical models of the therapy. The sample queries presented above are currently expressed in terms of KMeD's implementation of the PICQUERY+ tabular language [CIT+93].

3 Logical Data Model

In this section, the logical data model of the thermal ablation database is described. The model uses the KMeD framework of objects, image stacks, and type abstraction hierarchies to represent the data generated by the application domain [JC88, CCT95].

3.1 Patients and Procedures

The standard KMeD representation of a patient is shown in Figure 1. This representation stores an individual `Patient` as an entity participating in the relationship *undergoes* in conjunction with a `Procedure` entity. This relationship is 1-to-n; thus, a `Patient` undergoes one or more `Procedures`. A 1-to-1 relationship going in the other direction also shows that any given Procedure *is performed on* one and only one `Patient`.

The database keeps track of two types of procedures: `Examinations` and `Treatments`. `Examinations` typically generate a set of MRI (magnetic resonance imaging) scans of the patient's brain and tumor, including a measurement of the tumor's volume during that `Examination`. `Treatments` include further information, relating to the actual ablation procedure: the number of doses applied, whether or not a biopsy was taken, etc.

Within the context of the thermal ablation application, `Procedures` (both `Examinations` and `Treatments`) *study* one or more `Neoplasm` entities. Our modeling of a `Neoplasm` entity is described in the next section.

3.2 Neoplasms

The modeling of brain tumor, or `Neoplasm`, entities is needed to fulfill a number of requirements, the most significant of which are the ability to track a tumor's condition over time and the ability to capture the spatial features of a tumor. Our `Neoplasm` modeling also aims to fulfill other, more advanced requirements: in particular, the constructs described in this section will also be capable of supporting tumor simulation models and multimedia visualizations of these models or their real-world counterparts.

As can be seen in Figure 1, the `Neoplasm` entity is expressed in a different notation that represents a `Neoplasm` as a `Stream` entity. `Stream` entities are based on models and file formats defined in the area of multimedia and digital video [GBT93, SSJ93, App93, Gal91]. Our work extends this construct to a more general set of applications.

Stream Modeling A `Stream` entity represents a real-world object that changes over time. `Stream` entities can be viewed as a series of snapshots or states of the object being modeled, each series or state representing the object at a particular point in time.

Because one of the key requirements for the database was to track changes in a brain tumor's volume over time, the `Neoplasm` entity is modeled as a kind of `Stream` entity, where each snapshot within the `Stream` represents the condition (i.e. volume, position, appearance) of the `Neoplasm` during a particular point in time.

3.3 Images

Data modeling of images has long been a focus of the research with our multidisciplinary group [CCK81, JC88]. For the thermal ablation database, we extend the image stack construct introduced in [CCK81] and further developed in [JC88, CIT+93, CITB92] and [CCT95].

The image stack construct is an aggregation of related images along a third dimension. The image stack data model maps very well to the volumetric nature of MR data, and is therefore used to represent MR scans. These scans are instances of an image class called `PACSImage`, named after the Picture Archiving and Communications System at the UCLA Department of Radiological Sciences **[PACS refs]**.

The link between `PACSImages` and `Neoplasms` is a pair of relationships: `Neoplasms` *are represented in* `PACSImages`, and conversely, `PACSImages` *contain representations of* `Neoplasms`. These relationships permit the user to easily express queries of the forms "show me the image(s) where this neoplasm appears" and "show me the neoplasm(s) represented in this image."

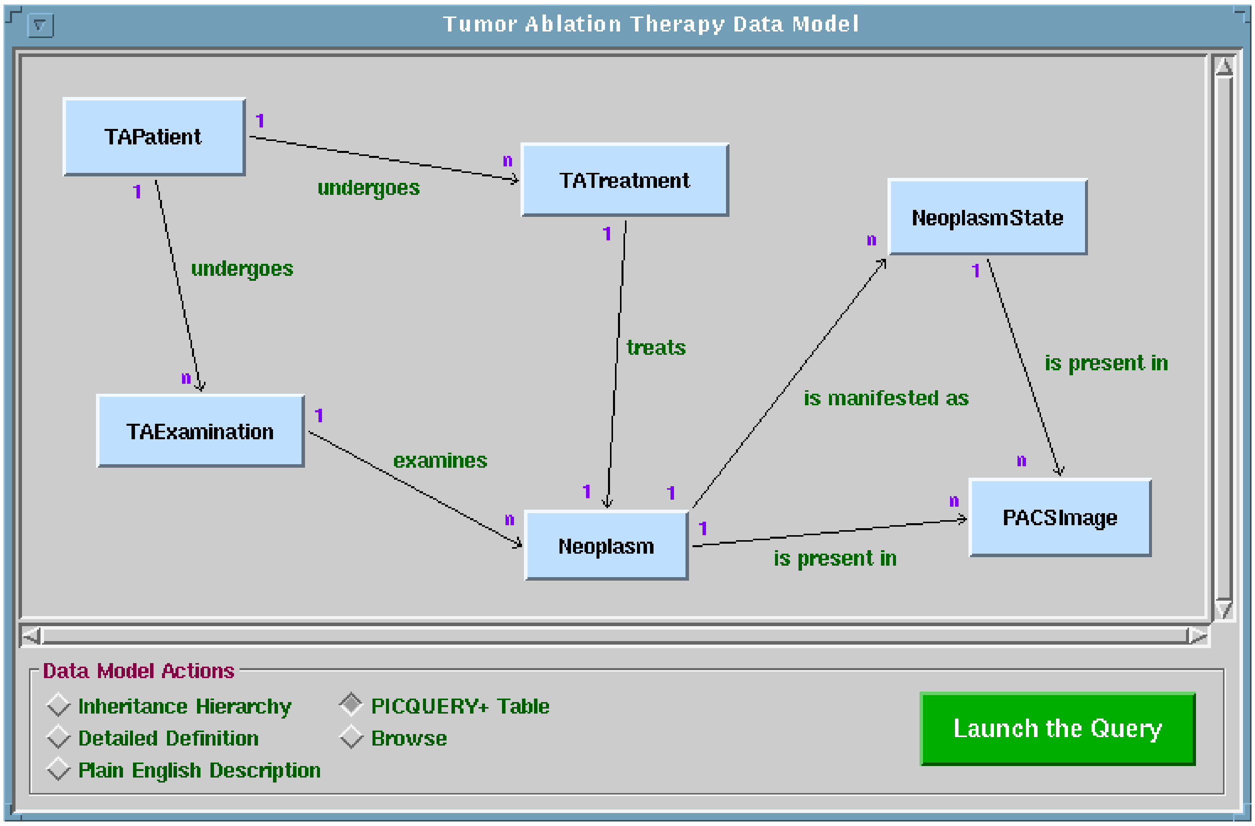

Figure 2: Data model display of the thermal ablation therapy query interface.

4 Query Interface

The interface developed for the thermal ablation database represents an intermediate step between our PICQUERY+ language [CIT$^+$93] and a more visually-oriented interface that is currently being developed. In this section, we describe the interface, along with a walk-through of a sample query.

4.1 Data Model Display

The visual portion of the thermal ablation therapy database is a representation of the system's data model, as shown in Figure 2. This diagram corresponds to the logical model shown in Figure 1, and should, ideally, match it exactly.

The primary difference between the implemented interface and the logical, paper diagram is the representation of a `Neoplasm`. A supplementary entity called the `NeoplasmState` is used to capture the features of the `Neoplasm` over time, and we say that a `Neoplasm` *is manifested as* a `NeoplasmState`. Full user interface implementation for the `Stream` construct (where this separation of static and time-varying components is no longer necessary) is currently in progress.

The data model display permits a number of operations, selected by clicking on the appropriate radio button at the bottom of the window. When an entity box is clicked (such as the `TAPatient` box), the corresponding operation is performed on that entity:

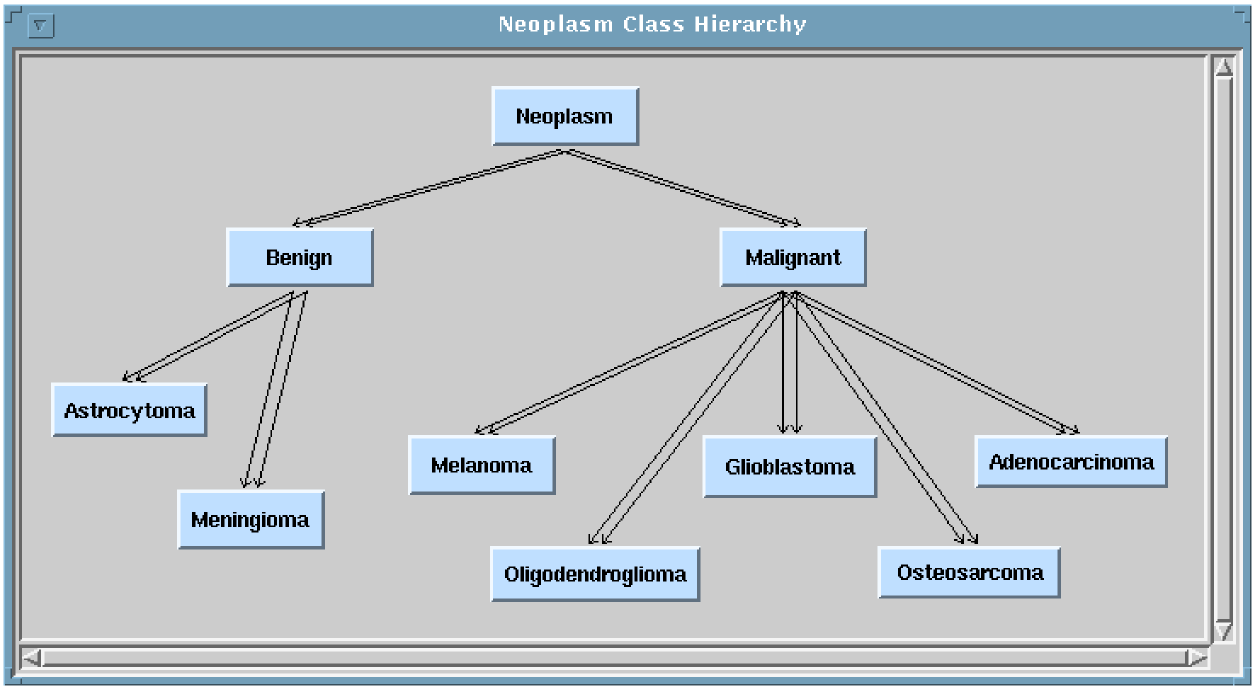

Figure 3: Type abstraction or inheritance hierarchy for the `Neoplasm` entity.

- *Inheritance hierarchy.* This operation displays the classification hierarchy to which the clicked entity belongs. Figure 3 illustrates the inheritance hierarchy that is displayed when the `Neoplasm` entity is clicked.
- *Detailed definition.* When a detailed definition is requested, a window displaying the internal structure of each entity is presented, including its attributes, attribute names, etc. Figure 4 illustrates the detailed definition of a `Neoplasm` entity, shown here to include the attributes patientID, location, and histology.
- *Plain English description.* A plain English description is a free text string that describes an entity in simple, non-technical English. These descriptions are stored in and retrieved from the KMeD global directory, along with the actual entries for the entities themselves. Clicking on an entity while this operation is active brings up a small text window containing that description. Figure 5 illustrates plain English windows for the `TAPatient` and `Neoplasm` entities.
- *PICQUERY+ table.* Clicking on an entity with the PICQUERY+ table operation selected activates the PICQUERY+ portion of the thermal ablation interface. A PICQUERY+ table on the selected entity (`Neoplasm` in Figure 6) appears, and the user can then type in predicates for filtering out the desired database objects. Details on PICQUERY+ are discussed in the next section.
- *Browse.* To access the database without issuing any queries or constraints, the user can choose the Browse operation then click on an entity. This action brings up a list of the actual data for the selected entity. Selections can then be made from this list for individual viewing by the user.

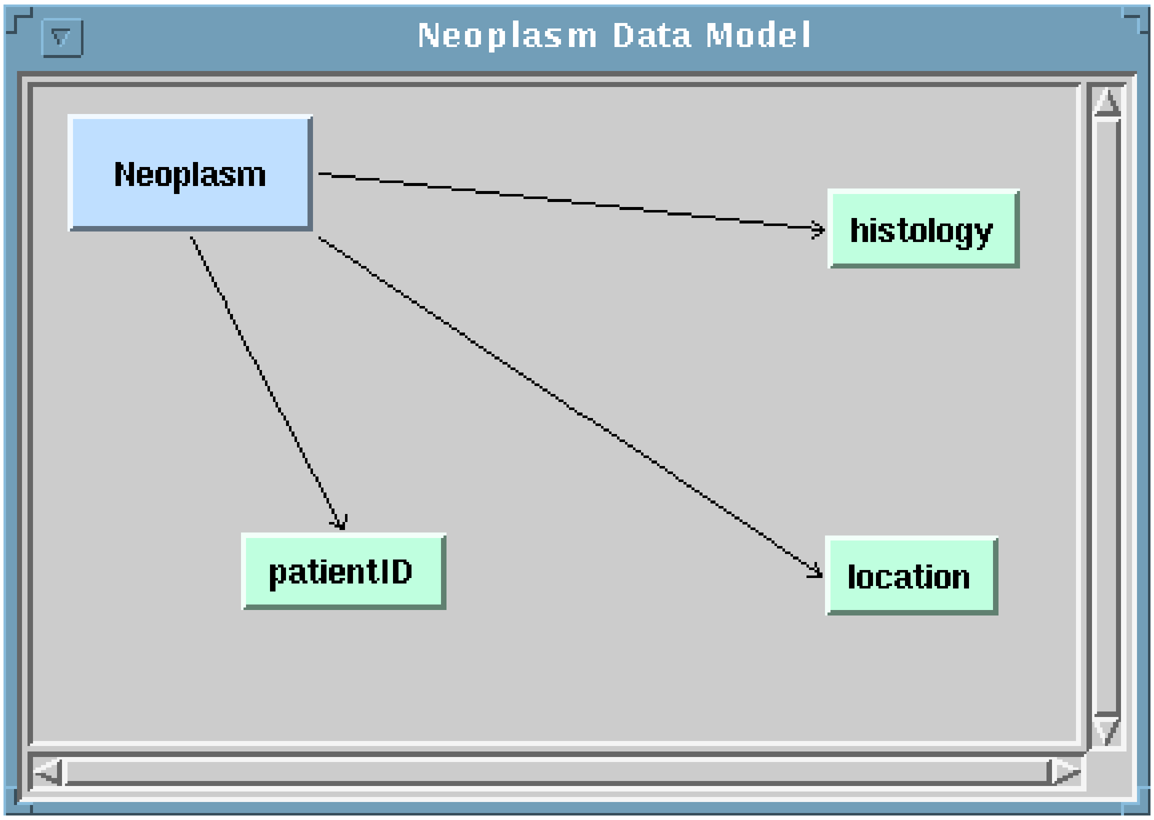

Figure 4: Detailed definition display for the `Neoplasm` entity.

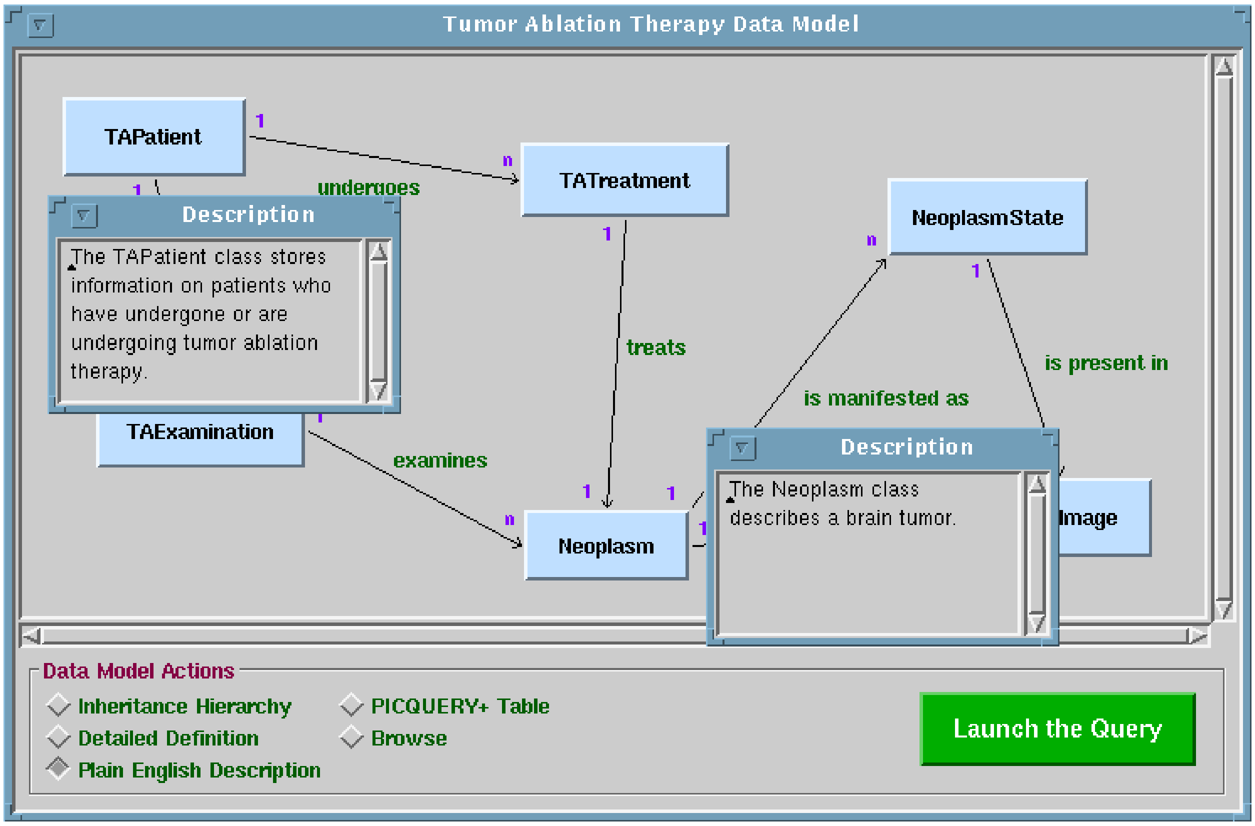

Figure 5: Plain English description windows for the `TAPatient` and `Neoplasm` entities (shown next to the entities that were clicked to invoke them).

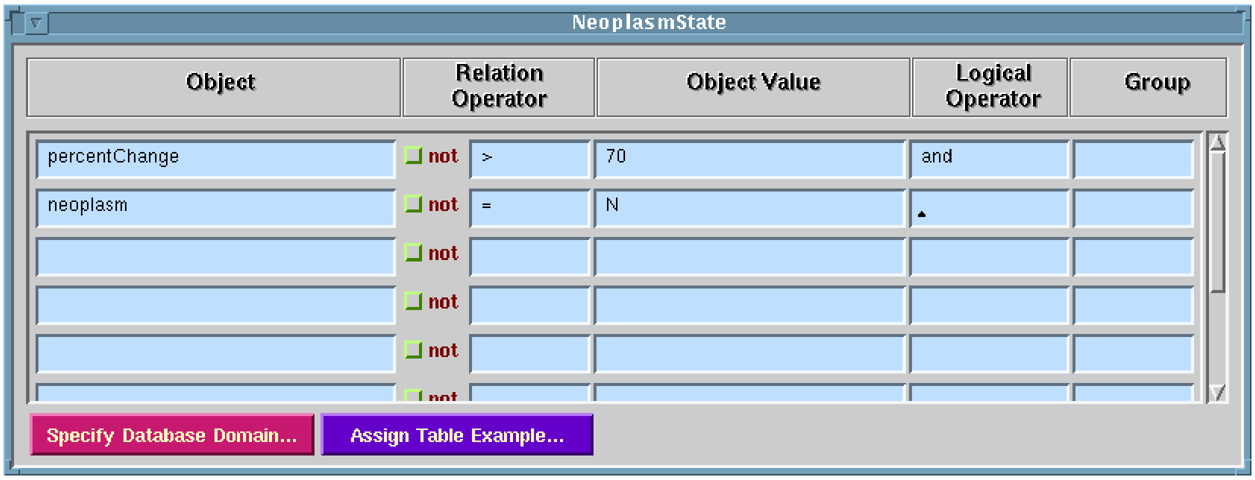

Figure 6: KMeD PICQUERY+ table that opens when the `NeoplasmState` entity is clicked during the PICQUERY+ table operation. The table has been filled with a single predicate requesting `NeoplasmStates` that have grown by at least 70% over the tumor's known history, with a link to their associated `Neoplasm` entities.

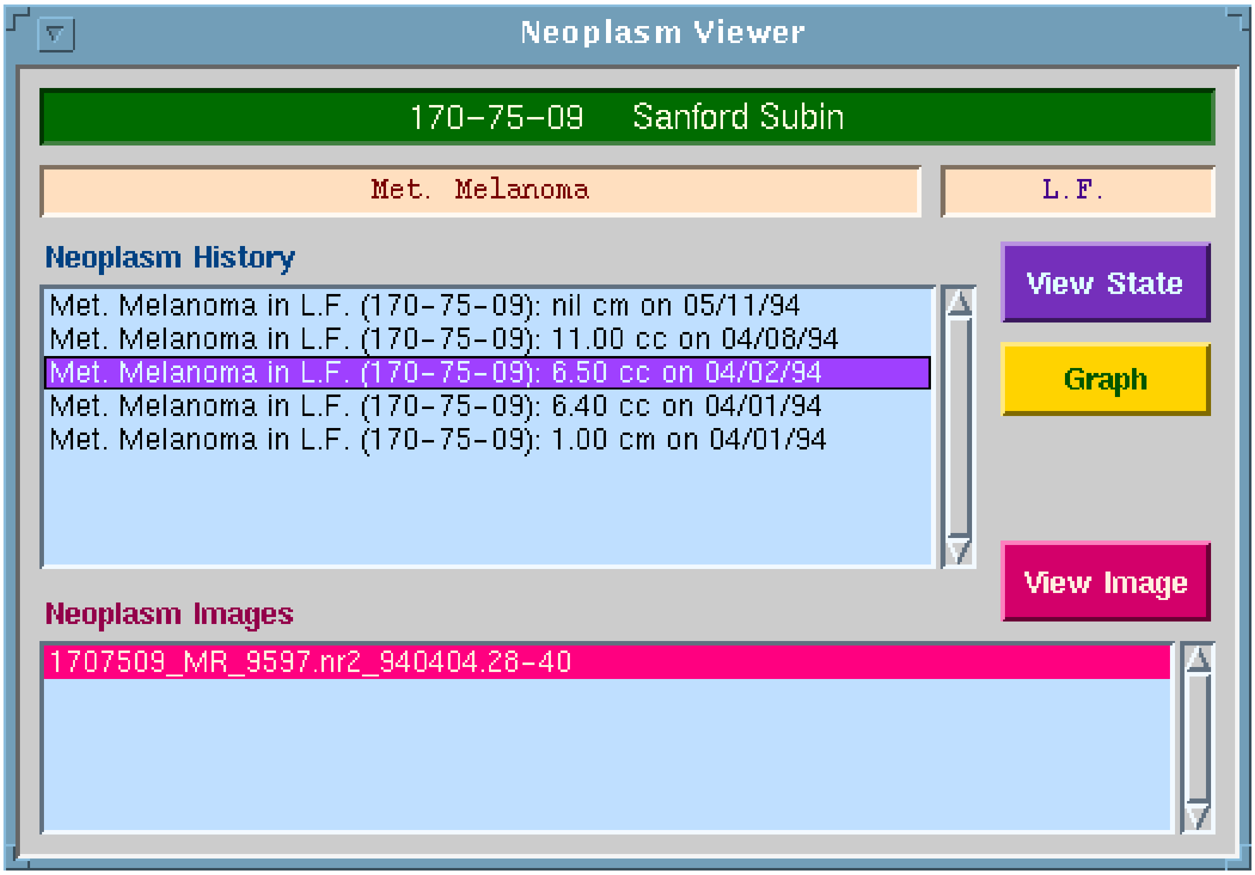

Figure 7: Display window for viewing Neoplasm entities, showing the list of hypertext-style links to associated images in the database.

4.2 PICQUERY+

When the user clicks on an entity while the PICQUERY+ table radio button is selected, a PICQUERY+ table on the selected entity appears. Using this table, the user can specify constraints or predicates on the given entity. In addition, multiple tables on multiple entities can be linked together (as in Query-By-Example [Zlo75]) to form more complicated queries. The best way to illustrate the language is by example, which will be presented in Section 4.4.

4.3 Hypertext-Style Links

When an object from the database has been retrieved and displayed onscreen, the user has the option of following links from that object to other entities in the database, if a relationship exists between those objects. For example, in Figure 7, a display window for a particular `Neoplasm` entity is shown. At the bottom of this window is a list of image files, representing the MR scans in which the displayed `Neoplasm` can be found. By clicking on the desired image then clicking on the *View Image* button, the user can follow that link to bring up the selected image.

The `Neoplasm` entity also provides links to the various states in which it was measured. By clicking on one of the items in the middle list then clicking on the *View State* button, the user can examine the `Neoplasm` as it was on the given date, with time-dependent information such as volume, percent change since the previous examination, etc.

4.4 A Sample Query Walk-Through

In this section, we illustrate how the query interface elements presented in the previous section are used and combined in order to answer one of the target queries. The query to be answered is:

> Which tumors in the database classified as "adenocarcinoma" grew by more than 70% after thermal ablation treatment?

We begin by opening the thermal ablation data model display shown in Figure 2. This is accessed from KMeD's main menu, which lists the available custom applications that use KMeD's database machinery.

The query to be answered needs to find the `Neoplasm` entities that have been given a histology classification of "adenocarcinoma" and that grew by more than 70% at least once after thermal ablation therapy. The initial histology classification, once assigned, no longer changes; thus, it can be found in the main `Neoplasm` entity, and results in the opening of a PICQUERY+ table for `Neoplasm`. In addition, volume change is an attribute that depends on time, and is therefore stored within the stream of `NeoplasmState`s that make up the entire history of the `Neoplasm`. Thus, a PICQUERY+ table on `NeoplasmState` must be opened.

The user clicks on the PICQUERY+ table data model operation, thus highlighting that radio button as the current one. Then, the `Neoplasm` entity is clicked. This brings up a PICQUERY+ table for `Neoplasms`. The same action is performed for `NeoplasmState` entities.

At this point, the query can either be typed in directly or entered with the assistance of menus. Each row corresponds to a predicate or filter by which to determine the data for retrieval. The various elements of each predicate are filled into the corresponding columns, and menus are context-sensitive to the column over which they are invoked.

To open the menu associated with a PICQUERY+ column, the user holds down the middle button over the appropriate query field. Since we are asking about a `Neoplasm`'s histology classification, we hold down the middle button over the first row of the *Object* column of the `Neoplasm` table. The *choose* submenu lists the attributes or functions that are applicable to the table's underlying entity; for this query, the attribute histology is selected as the object of interest. A user who knows the database well may also type the term 'histology' directly into the table's cell. We then define that this histology attribute must contain the text "Adenocarcinoma" as its value.

Following a similar procedure of either using pull-down menus or typing directly, the following predicates are then filled into the `NeoplasmState` table:

```
percentChange   >   70   and
neoplasm       =   N
```

The second line is necessary because we must define the link between the `Neoplasm` and `NeoplasmState` tables. Here, we express the link by using the example element "N" to stand for the `Neoplasm` being referred to by the `NeoplasmState`.

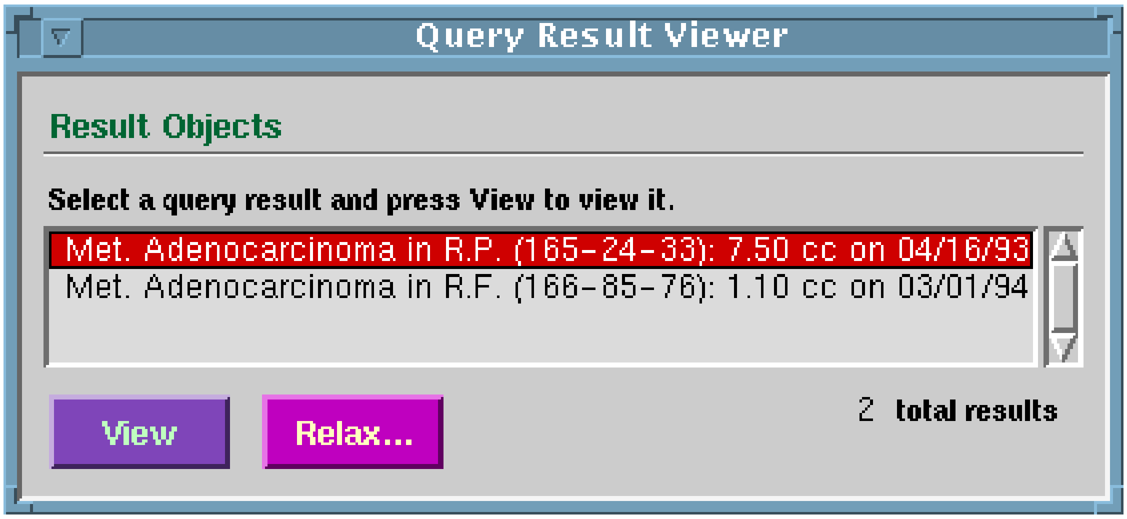

Figure 8: Query results returned by KMeD for our sample query on the thermal ablation database.

Using a mechanism similar to example elements in Query-By-Example, the user assigns this example element 'N' to the `Neoplasm` table by clicking on *Assign Table Example* and entering 'N' when asked for an example name. Thus, the `neoplasm = N` predicate in the `NeoplasmState` table effectively links the `NeoplasmState` to its corresponding `Neoplasm`. Note also that if the user wishes to further restrict the `Neoplasms` to be retrieved (i.e. select only `Neoplasms` located in the right frontal region of the brain, etc.), then the desired predicates may be entered into the `Neoplasm` table too.

This completes the formation of the PICQUERY+ query, which can be seen in Figure 6. The user now clicks on the *Launch The Query* button to initiate query processing. After a few moments, KMeD returns with a list of available results, as shown in Figure 8.

4.5 Query Relaxation

As can be seen in Figure 8, only a small number of neoplasms satisfied the original query. If desired, the user may *relax* the query in order to increase the number of retrieved results. The relaxation operation is carried out by the generalization operation (moving up the type abstraction hierarchy) [CC94]. For our sample query, the `Adenocarcinoma` is generalized to a `Neoplasm`, which covers various kinds of brain tumors (Figure 3).

To relax the query, the user clicks on the *Relax...* button in the query result view. This opens up the KMeD Relaxation Manager (Figure 9), which lists the query predicates that may be relaxed. By relaxing the "histology contains 'Adenocarcinoma'" predicate, the neoplasm's type abstraction hierarchy is consulted to retrieve other tumors that are related to the adenocarcinoma classification. Clicking on the *Relax Now* button then expands the predicate accordingly.

When the *Requery* button is clicked, the query is re-executed with the relaxed predicate. Figure 10 shows the new set of results, which now lists an increased number

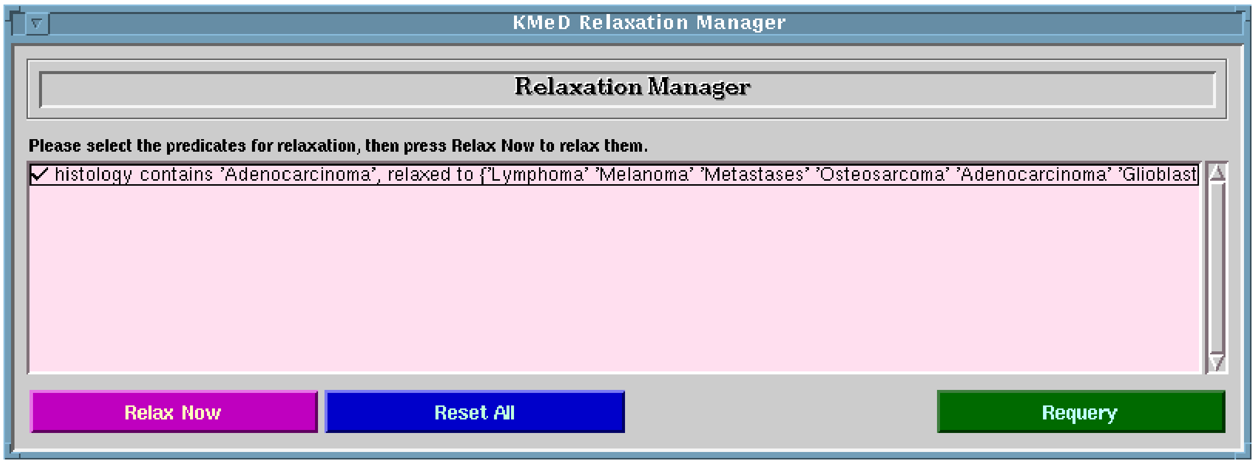

Figure 9: Query relaxation parameters as defined through the KMeD Relaxation Manager.

of retrieved tumors.

5 Output Visualization

The final phase in the interaction with the thermal ablation database, whether with the sample query shown previously or with any other query, is the display and visualization of query results. Previous sections have already illustrated some of the data display capabilities of KMeD and the thermal ablation database; in this section, these facilities are described and illustrated more fully.

5.1 General Approach

Figure 7 illustrated KMeD's approach to displaying retrieved data: each KMeD entity is given one or more custom display windows tailored to the attributes and relationships of that entity. If more than one display type is available for an entity, a menu automatically opens up, listing the available display approaches. The window shown in Figure 7 is the layout that was designed for Neoplasm entities. Other types of entities, such as Patients, Examinations, Treatments, etc., may also be given such layouts. If the user does not provide a custom window design, KMeD generates one on-the-fly, though the on-the-fly layout tends not to be very useful except for basic examination of the entity's attributes.

This modular approach is particularly useful for multimedia data; in this case, we are dealing with MR images. Because images may be visualized in many different ways, and such visualization methods vary depending on the application domain, it is beneficial to have the capability to plug and unplug new visualization modules as application needs evolve.

Succeeding sections describe the visualization facilities that are currently available on KMeD and the thermal ablation database. Keep in mind that these options are not locked into KMeD; depending on needs or resources, virtually any visualization module may be connected to KMeD, as long as image data can be sent to the visualization module.

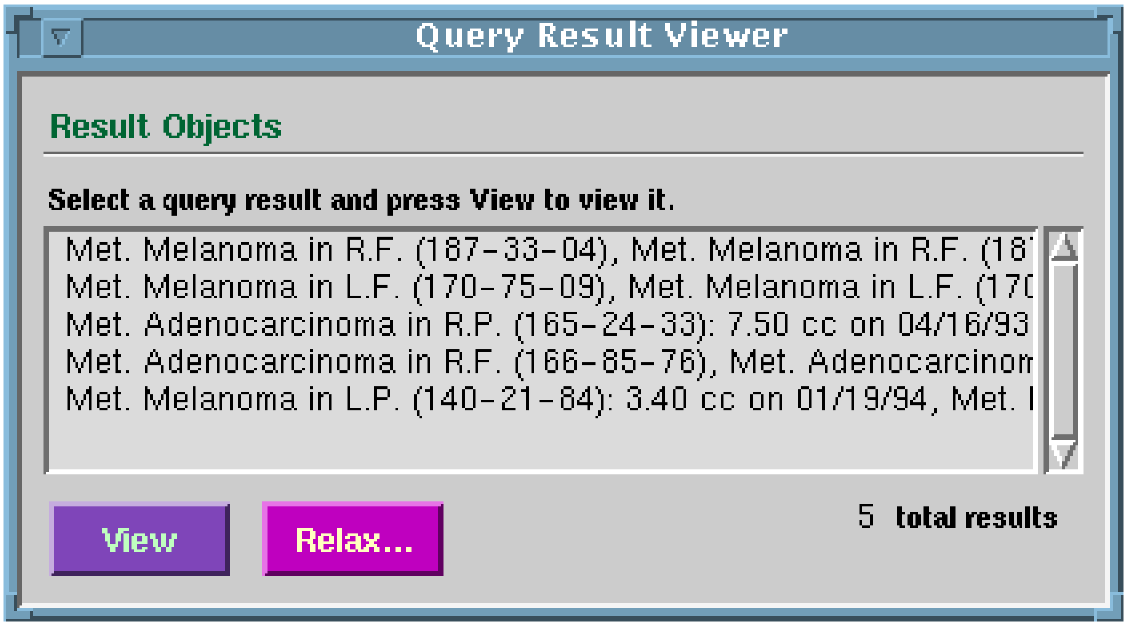

Figure 10: Query result window for a new, relaxed version of our sample query. In English, the relaxed version roughly translates to "Which tumors whose classification is similar to 'adenocarcinoma' have grown by more than 70% at least once during their known history?"

5.2 Tabular and Graphical Output

One of the most important factors in thermal ablation therapy is the changes in size that the tumor undergoes through time. This change in size is best visualized by a graph viewer, shown in Figure 11. The graph viewer retrieves all of the states associated with a Neoplasm entity, sorts them by date, then plots their volumes in a window. The graph view provides the user with an immediate feel for the relative success of the thermal ablation therapy.

5.3 Single Image Visualization

Most of the image visualization modules in KMeD are capable of viewing a single image or slice from the MR scan. A particularly useful tool is the `display` module from the ImageMagick package. which is available in the public domain. The `display` module permits the user to view one MR slice at a time, with a "hot magnification" feature that provides a real-time 2× view which follows the movement of the mouse over the normal-sized image. Figure 12 illustrates this real-time magnification display.

5.4 Cine-Style Visualization

Another mode of visualization for image data that consists of multiple slices is *cine mode*, or the ability to "step" through each slice of an image stack either one by one or in an animation sequence. We have a number of modules that support cine mode, including ImageMagick's animate program and software that was developed by our group using the IDL (Interactive Data Language) [Res94] system and VisualWorks 1.0

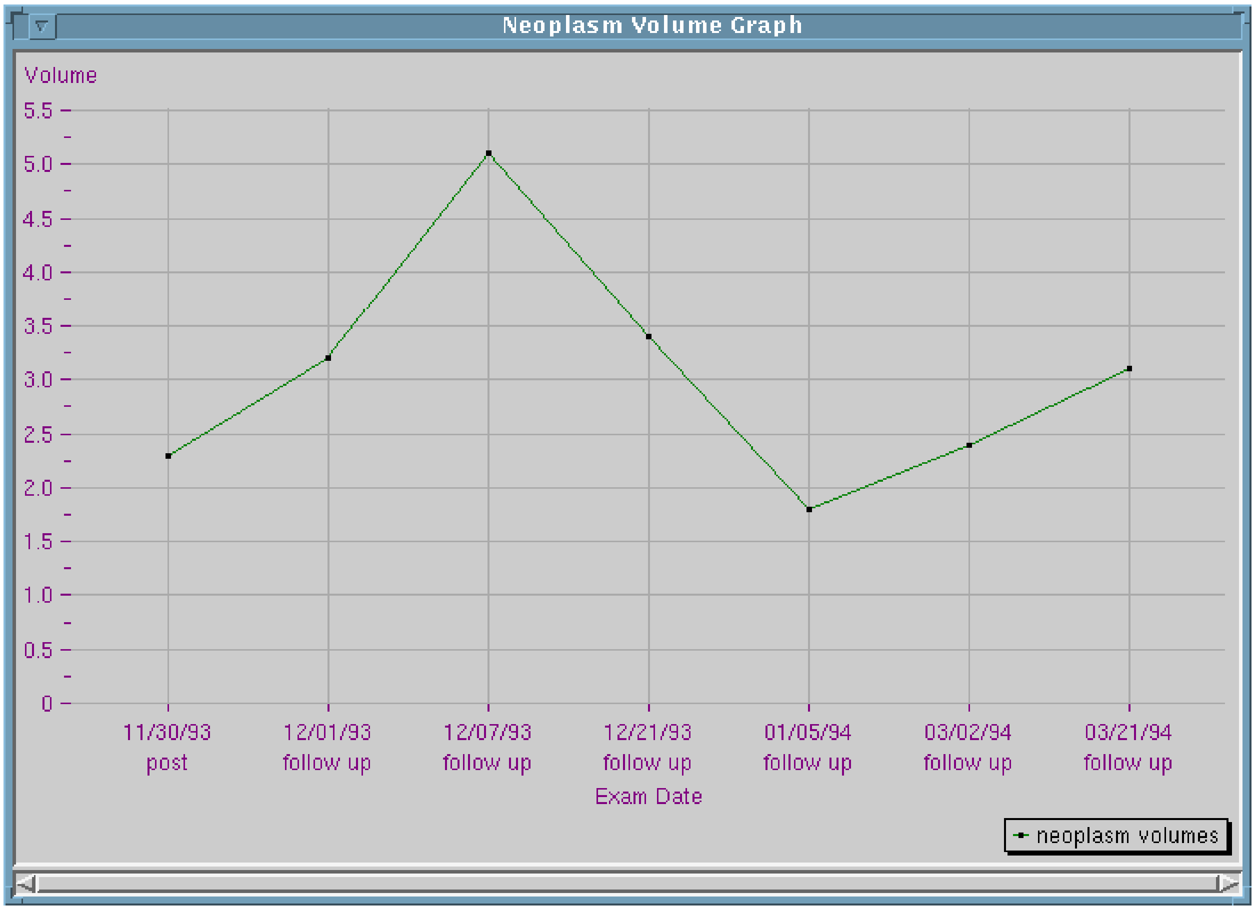

Figure 11: Graph window used for visualizing the volume change in a tumor over time.

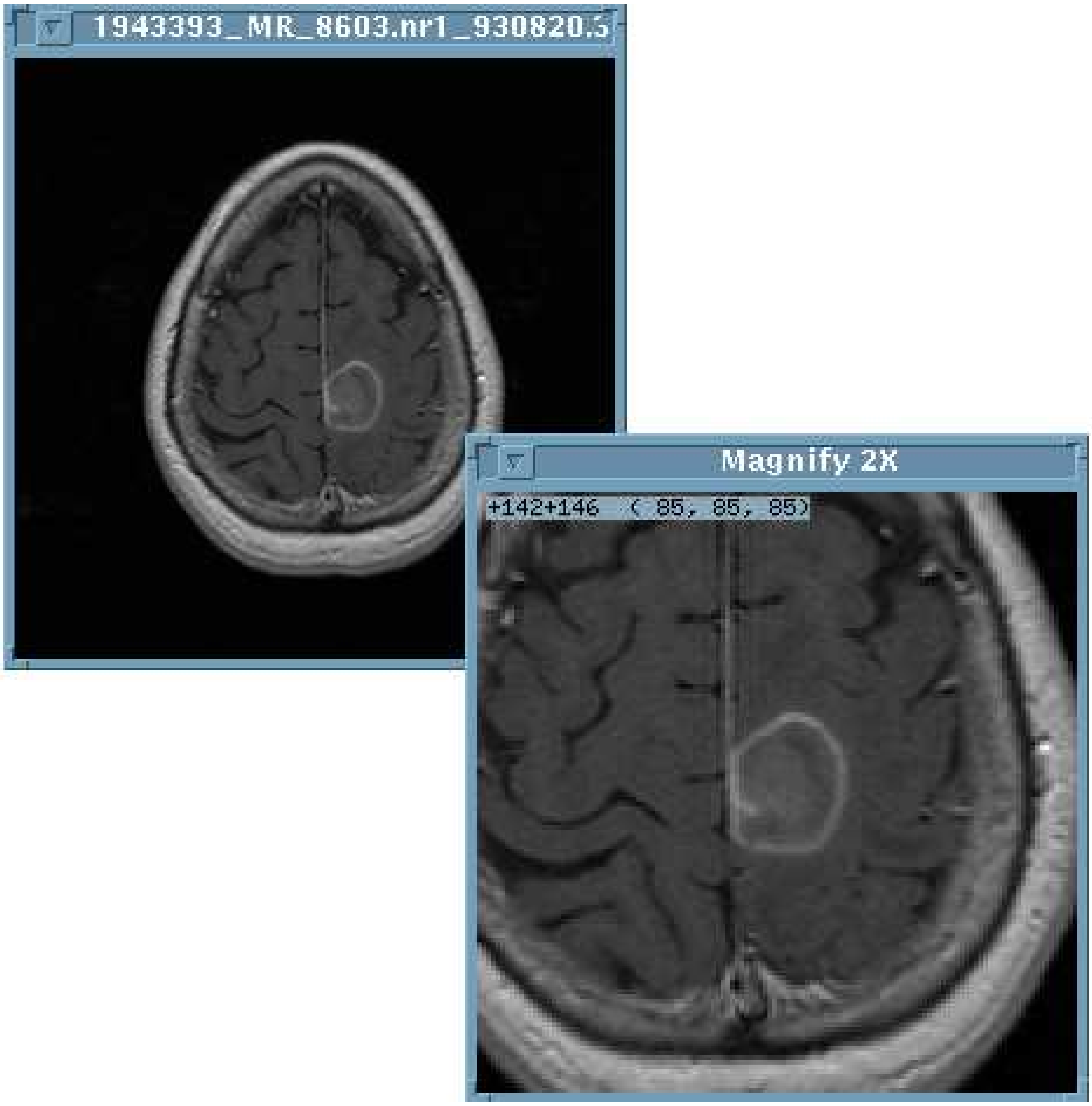

Figure 12: ImageMagick's `display` module with its real-time magnification feature. The magnified window follows the movement of the mouse.

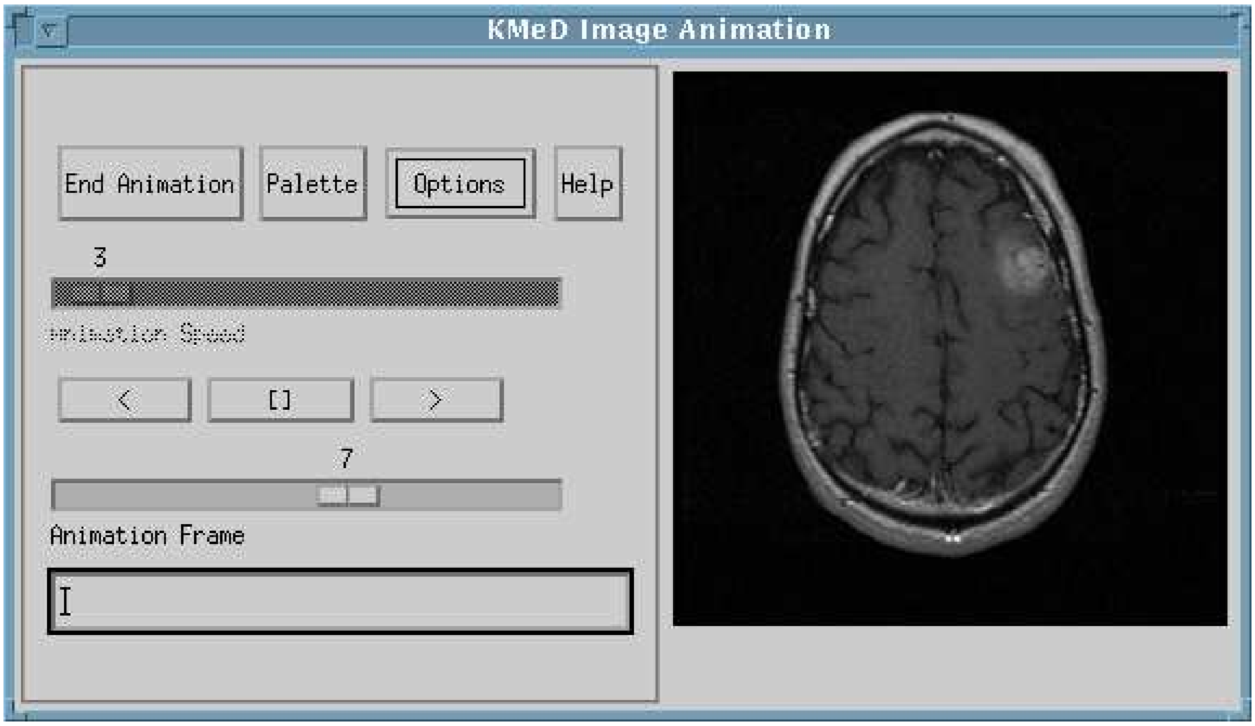

Figure 13: Cine-style visualization module as implemented in IDL. The brain tumor can be seed as the light-colored nodule in the right frontal region of the brain.

[Par92]. Figure 13 illustrates the cine mode visualizer that was implemented using IDL.

5.5 3D Visualization

MR data is volumetric data, and so can be meaningfully viewed in three dimensions. We adapted IDL's `slicer` program so that it could communicate with KMeD; with the slicer, an MR scan may be viewed along any 3D axis, rendered as an isosurface, or cut into three-dimensional blocks. Rotations and 2D plane slices on any angle are also permitted. Figure 14 illustrates the KMeD slicer window.

6 Conclusions

MR guided thermal ablative therapy of brain tumors is a revolutionary investigational technique that has potential advantages over other treatments. A conveniently accessible database is essential in order to properly evaluate the strengths and weaknesses of the new technique.

The database described in this document is built as a high-level application running on top of the KMeD project which our group has been developing since 1991. Significant features of the database include the ability to query the tumor data through meaningful features such as tumor volume, the ability to track the development of a tumor over time, and the ability to retrieve and visualize MR images of a patient with a given tumor.

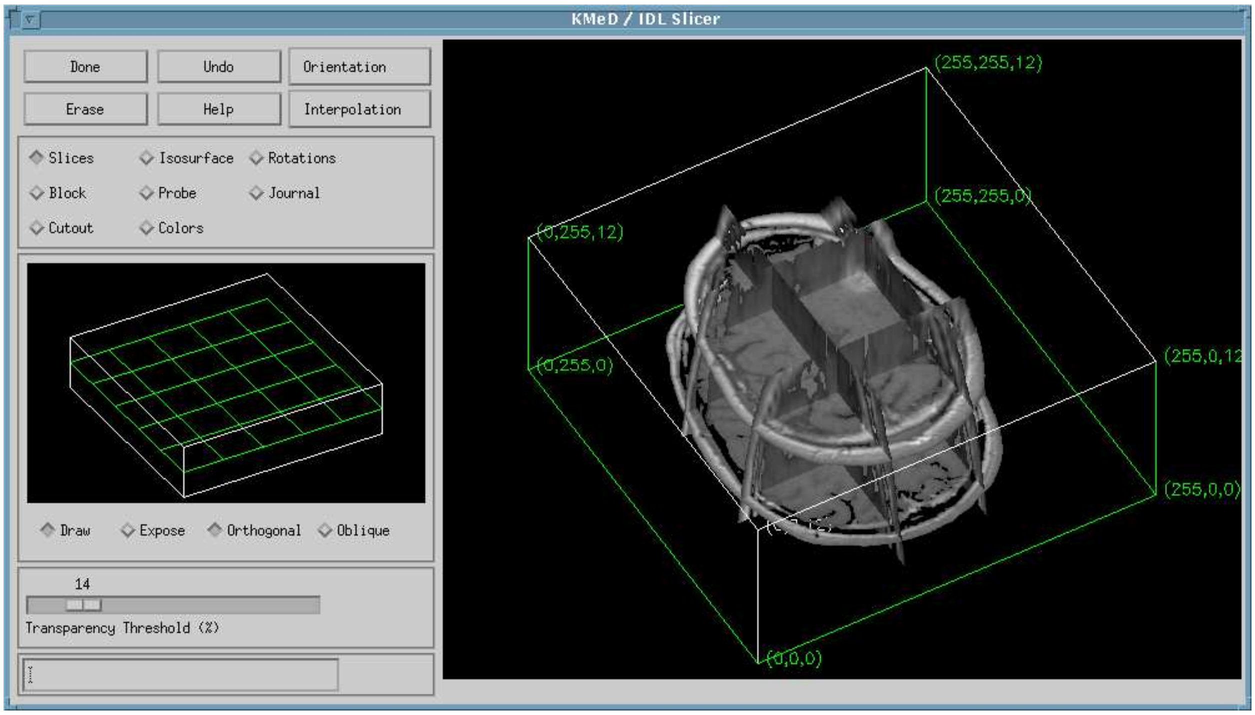

Figure 14: 3D slice-by-slice view of an MR scan using a slicer program implemented in IDL.

The thermal ablation database presented here, including its data model, query interface, and visualization facilities, represents what has been implemented as a real system to date. Our multidisciplinary group intends to build further on these results by working on full implementation of the Stream model, development of more sophisticated visualization and annotation techniques, and addition of simulation modeling and validation to the overall information structure.

References

[ADB+93] Y. Anzai, A. DeSalles, K. Black, K. Farahani, S. Sinha, D. Castro, and R. Lufkin. Interventional MRI. *Radiographics*, 1993.

[Aea91] Y. Anzai and R. Lufkin et. al. MRI guided interstitial Nd:YAG laser phototherapy: A dosimetry study of acute tissue damage in the in vivo model. *Journal of Magnetic Resonance Imaging*, 1(5):553–559, September–October 1991.

[ALD+95] Y. Anzai, R. Lufkin, A. DeSalles, D. R. Hamilton, K. Farahani, and K. L. Black. Preliminary experience with MR-guided thermal ablation of brain tumors. *American Journal of Neuroradiology*, 16(1):39–48, January 1995. Discussion on pp. 49–52.

[App93] Apple Computer, Inc. *QuickTime*. Inside Macintosh. Addison-Wesley Publishing Company, 1993.

[Arr60] S. Arrow. The use of radiofrequency power in making lesions in the brain. *Journal of Neurosurgery*, 17:431–438, 1960.

[CC94] Wesley W. Chu and Qiming Chen. A structured approached for cooperative query answering. *IEEE Transactions on Knowledge and Data Engineering*, 6(5):738–749, October 1994.

[CCK81] Margaret Chock, Alfonso F. Cardenas, and Allen Klinger. Manipulating data structures in pictorial information systems. *Computer*, pages 43–50, November 1981.

[CCT95] Wesley W. Chu, Alfonso F. Cardenas, and Ricky K. Taira. KMeD: A knowledge-based multimedia medical distributed database system. *Information Systems*, 1995. To appear.

[CIT+93] Alfonso F. Cardenas, I. T. Ieong, R. K. Taira, R. Barker, and C. M. Breant. The knowledge-based object-oriented PICQUERY+ language. *IEEE Transactions on Knowledge and Data Engineering*, 5(4):644–657, August 1993.

[CITB92] W. W. Chu, I. T. Ieong, R. K. Taira, and C. M. Breant. A temporal evolutionary object-oriented data model and its query language for medical image management. In Li-Yan Yuan, editor, *Proceedings of the 18th International Conference on Very Large Databases*, pages 53–64, Vancouver, Canada, August 1992. Very Large Data Base Endowment, Morgan Kaufmann Publishers, Inc.

[CSL92] D. Castro, R. E. Saxton, and R. Lufkin. Interstitial photoablative laser therapy guided by magnetic resonance imaging for the treatment of deep tumors. *Seminars of Surgical Oncology*, 8:233–241, 1992.

[Gal91] Didier Le Gall. MPEG: A video compression standard for multimedia applications. *Communications of the ACM*, 34(4):46–58, April 1991.

[GBT93] Simon Gibbs, Christian Breiteneder, and Dennis Tsichritzis. Data modeling of time-based media. In Dennis Tsichritzis, editor, *Visual Objects*, pages 1–22. Centre Universitaire d'Informatique, Université de Genève, 1993.

[JC88] Thomas Joseph and Alfonso F. Cardenas. PICQUERY: A high level query language for pictorial database management. *IEEE Transactions on Software Engineering*, 14(5):630–638, May 1988.

[Par92] ParcPlace Systems, Inc., Sunnyvale, CA. *VisualWorks Release 1.0 User's Guide*, 1992.

[Res94] Research Systems, Inc. *IDL 3.5 User's Guide*, 1994.

[SMH60] W. H. Sweet, V. H. Mark, and H. Hamlin. Radiofrequency lesions in the central nervous system of man and cat: Including case reports of eight bulbar pain-tract interruptions. *Journal of Neurosurgery*, 17:213–225, 1960.

[SSJ93] Deborah Swanberg, Chiao-Fe Shu, and Ramesh Jain. Knowledge guided parsing in video databases. In Keith T. Knox and Edward Granger, editors, *IS&T/SPIE's Symposium on Electronic Imaging: Science & Technology*, San Jose, California, USA, January–February 1993. The Society for Imaging Science and Technology (IS&T) and The International Society for Optical Engineering (SPIE).

[Zer65] N. T. Zervas. Eccentric radiofrequency lesions. *Confin Neurol*, 26:143–145, 1965.

[ZK72] N. T. Zervas and A. Kuwayama. Pathological characteristics of experimental thermal lesions: Comparison of induction heating and radiofrequency electrocoagulation. *Journal of Neurosurgery*, 37:418–422, 1972.

[Zlo75] M. M. Zloof. Query-by-example. In *Proceedings of the National Computer Conference*, pages 431–437, Arlington, VA, May 1975.

[ZWL+93] J. Zhang, C. L. Wilson, M. F. Levesque, E. J. Behnke, and R. B. Lufkin. Temperature changes in nickel-chormium intracranial depth electrodes during MR scanning. *American Journal of Neuroradiology*, 14(2):497–500, March–April 1993.