

AD-A252 084 ON PAGE

Form Approved
OMB No. 0704-0188Public release
gathering
collection
Davis Hqg

1 hour per response, including the time for reviewing instructions, searching existing data sources, collection of information. Send comments regarding this burden estimate or any other aspect of this reporting requirement to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Avenue, Washington, DC 20503.

1. AGI

April 30, 1992

3. REPORT TYPE AND DATES COVERED

Interim Tech. Report, July 1990- Sept. 1991

4. TITLE AND SUBTITLE

An Iconic Query Interface for an Image Database

5. FUNDING NUMBERS

DAAL03-89-G-0118 ✓

6. AUTHOR(S)

V. N. Gudivada and V. V. Raghavan

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Jackson State University, Jackson, MS 39217
University of SW Louisiana, Lafayette, LA 70504JUN 12 1992
PERFORMING ORGANIZATION
REPORT NUMBER

91-2

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U. S. Army Research Office
P. O. Box 12211
Research Triangle Park, NC 27709-221110. SPONSORING/MONITORING
AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Recently, there has been widespread interest in various kinds of database management systems for managing formatted information. Powerful and flexible querying mechanisms are provided as an integral part of any of the conventional database management systems for formatted data so that they can be highly useful to a casual database user. Querying schemes assume even more critical prominence in systems used for image data management. Most of the existing schemes for querying image databases are variations of either Query-by-Example (QBE) or Structured Query Language (SQL). We feel that this approach to querying image databases is very unsatisfactory in terms of their expressive power and also being not natural to the image data.

In this report, we propose a unified multi-layered framework for retrieval in image databases. This framework is expected to be highly flexible enough so that it can be useful across several application areas. The system is implemented using Smalltalk/V Windows which runs under a Graphical User Interface (GUI) known as Microsoft Windows. The system enables a user to interactively sketch an iconic query and then computes its spatial similarity with each and every image in the collection. The images in the collection are then shown in the decreasing order of their similarity for a specified query.

14. SUBJECT TERMS

Iconic Queries, Similarity Based Retrieval, Image Databases

15. NUMBER OF PAGES

14

16. PRICE CODE

17. SECURITY CLASSIFICATION
OF REPORT

UNCLASSIFIED

18. SECURITY CLASSIFICATION

UNCLASSIFIED

19. SECURITY CLASSIFICATION
OF ABSTRACT

UNCLASSIFIED

20. LIMITATION OF ABSTRACT

UL

An Iconic Query Interface for an Image Database*

V. Naidu Gudivada and Vijay V. Raghavan

Technical Report: 91-2



Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

**Department of Computer Science
Jackson State University
Jackson, MS 39217**

* This research is supported by the Department of Defense under Grant No: DAAL03-89-G-0118.

92-15305



92 6 11 034

An Iconic Query Interface for an Image Database

Abstract

Recently, there has been widespread interest in various kinds of database management systems for managing formatted information. Depending upon the domain, these systems are variously referred to as Multimedia Databases, Spatial Databases, Pictorial Information Systems, and Image Database Systems. Powerful and flexible querying mechanisms are provided as an integral part of any of the conventional database management systems for formatted data so that they can be highly useful to a casual database user. Querying schemes assume even more critical prominence in systems used for image data management. Most of the existing schemes for querying image databases are variations of either Query-by-Example (QBE) or Structured Query Language (SQL). We feel that this approach to querying image databases is very unsatisfactory in terms of their expressive power and also being not natural to the image data.

In this report, we propose a unified multi-layered framework for retrieval in image databases. This framework is expected to be highly flexible enough so that it can be useful across several application areas. Only a part of the framework which addresses spatial relationships among objects in an image is implemented as of this writing. However, it should be noted that the proposed framework is very general and can address both spatial similarity and semantic similarity. To illustrate our scheme, we selected several residential floor plans as our image data. The system is implemented using Smalltalk/V Windows which runs under a Graphical User Interface (GUI) known as Microsoft Windows. The system enables a user to interactively sketch an iconic query and then computes its spatial similarity with each and every image in the collection. The images in the collection are then shown in the decreasing order of their similarity for a specified query.

1. Introduction

Recently, there has been widespread interest in various kinds of database management systems for managing formatted information. Depending upon the domain and the nature of formatted data, these systems are variously referred to as Multimedia Information Systems [33, 31, 34], Spatial Databases [18, 19], Pictorial Information Systems [3, 5, 6, 8], and Image Database Systems [7, 4, 21, 17]. The application areas for these systems include but not limited to Medical Imaging [40], Medical Information Systems [26, 27], Document Image Processing and Office Information Systems [15, 16, 29], Remote Sensing and Management of Earth Resources [14, 32], Geographic Information Systems and Cartographic Modeling [23, 43], Mapping, Land Information Systems [12], Robotics [24], Interactive Computer-Aided Design (CAD) and Computer-Integrated Manufacturing Systems (CAM) [24, 11], and Image Understanding Systems [28]. Though these application areas are diverse, they all view image data as a principal resource which needs to be integratedly managed with other types of data such as voice and conventional formatted data.

Tamura and Yokoya [41] provide an excellent survey of image database systems that were in practice around early 1980s. They classify image database systems into three categories. The classification is based on the data model employed, processing operations provided, and levels of abstraction in the image representation. Chock also [10] provides a good survey and comparison of functionalities of several image database systems for geographic applications. The functionality of image database systems ranged from simple cataloging used for distribution of remotely sensed imagery to image understanding and similarity measures required in medical imaging and medical information systems. In view of the recent developments in the field we classify the image database systems and approaches to implementation into five different categories. The five categories are:

1. Conventional Database Systems as Image Database Systems
2. Image Processing Systems Enhanced with Advanced File System/Database Functionality
3. Building Extensions and Extensibility into the Conventional Database Systems
4. General Purpose Image Database Systems
5. Ad hoc Approaches from Application Domains

Powerful and flexible querying mechanisms are provided as an integral part of the conventional database management systems for alphanumeric data so that they can be highly useful to a casual database user. Querying schemes assume even more critical prominence in systems used for image data management. Most of the existing schemes for querying image databases are variations of either Query-by-Example (QBE) or Structured Query Language (SQL). We feel that this approach to querying image databases is very unsatisfactory in terms of their expressive power and also being not natural to the image data.

In order to overcome some of the limitations of the existing query languages for image databases, in this report, we propose a unified multi-layered framework for retrieval in image databases. This framework is expected to be highly flexible enough so that it can be useful across several application areas. Only a part of the framework, which

addresses spatial relationships among objects in an image, is implemented as of this writing. The remainder of this report is organized as follows. Several query languages that have been proposed in the literature are briefly discussed in section two. We identify the limitations of the current query languages to retrieval in image databases and describe our proposed method in section three. Section four describes our prototype system implementation and section five provides conclusions and indicates future work.

2. Query Languages

The approaches to querying image databases that have been undertaken in the existing systems can be classified into the following three categories.

1. Extensions to the host database system query language.
2. Command Language designed to specifically suit the application requirements.
3. Logic-based query languages.

The *extended query language approach* is typically found in majority of the systems which are recently built on top of an existing conventional database system. The query languages associated with conventional databases in general perform unsatisfactorily except in case of data extracted from images and formatted in a manner essentially identical to those managed by conventional database systems. These systems employ fragmented image representation to suit the relational tabular format. It is quite natural to take advantage of host database system for querying the image data also. If the images are converted to symbolic representations and stored in relations, the only extension needed to the query language is the display management for image output. On the other hand, in systems where image data is stored in a separate database with spatial indexing, if the query language allows query specification in pictorial form, then the extensions needed may not be trivial. In general, a query may reference only the image data, only the non-image data, or a combination of image and non-image data. An ideal query language should provide a consistent user interface for both image and non-image data and be able to command and coordinate both alphanumeric query processor and the spatial query processor in a transparent way to the user. This approach is taken in PSQL [37] and in PICQUERY [22]. PSQL is an extension of SQL [2] and PICQUERY has a flavor similar to QBE [44] and QPE [4].

GEO-QUEL [1] was proposed as a query language for a geographical information system built on top of INGRES, which is the host system of QUEL. Geographic data (maps) can be manipulated by special GEO-QUEL commands or by using the query language QUEL. POSTQUEL is a recent extension to QUEL [38]. The extensions support complex objects, user defined data types and access methods, time varying data, iteration queries, alerters, triggers, and rules. SEQUOIA 2000 [39] project has adopted POSTGRES to be used for performing *global change research* involving very large spatial databases.

Query by a Pictorial Example (QPE) is the picture query language proposed for IMAID system [5]. This appears to be a misnomer since pictorial entities such as line segments are specified by the actual coordinate values of the end points. The system

provides several pictorial operators such as intersection coordinates of a line and a region and union of two regions to support spatial queries.

ADT-INGRES uses QUEL as a query language and both NF² data model and R²D² model use a query language similar to SQL. GEM is a general purpose query and update language for the entity relationship model and is designed as an extension of QUEL. Augmented SEQUEL is used in IDMS and ADM. Augmented relational algebra is used in GRAIN and augmented relational calculus is used in ARES [20]. Interestingly, augmented APL is used in AQL.

An example of *command language approach* to pictorial retrieval is found in GADS. IQ [25] is an interactive command driven image query language designed to support image editing, transformation, storage, retrieval and display. ELAS has command driven query interface where as ERDAS has a menu driven interface.

The KBGIS-II system uses a *logic-based query language* called spatial object language (SOL) [30]. The query language also allows for spatial constraint specification. Chang et al. [9] proposed a 2D G-string based query language. This query language processes queries using string matching and spatial reasoning.

User Interface requirements should be carefully evaluated and incorporated into the design of pictorial query languages. Eigenhofer and Frank [13] discuss user interface considerations in designing a pictorial/spatial query language. The query languages designed for conventional databases are clearly unsuitable for specifying pictorial queries. There are several types of spatial/image queries and each type may require a different method of specification. For example, a user may wish to indicate an area of interest on a map to search for a specific feature using an interactive input device such as a mouse. Users specify point queries to obtain information on all the objects that occupy a specified location in space. Region queries are used to obtain information on all the objects that exist in space enclosed by a hyper rectangle called query window. A query window can be conveniently specified in two dimensions by indicating two points using a pointing device.

The point and region queries can be combined with SQL queries to increase the expressive power of a query language. For example, a query can be specified to select only those objects within the query window that satisfy given non-spatial predicates. This type of query specification is highly suitable for applications dealing with geographic data. However, for querying image databases of electronic product catalogues, browsing techniques are effective.

In applications involving retrieval based on similarity, iconic query interface may be essential. The user specifies a query by placing icons designating real objects in the domain at certain desired locations in a query window and then assigns attribute properties to each of these icons. By doing so, the user is able to specify the spatial relationships among domain objects as well as their attribute values in the query. Retrieval based on "conceptual" similarity between images may require yet another method of query specification. If the response to a query involves the display of some images, the query language must in addition provide for the specification of an appropriate output device of user's choice. These problems compound and make the design of a pictorial query language a difficult task. Many of the current approaches to querying image databases address only a few aspects of this complex task, mostly driven by a specific application needs. Our goal is to provide different types of query

specification schemes that are most natural and integrated under a unified framework using a consistent user interface. We describe our approach to retrieval of images in a general purpose image database system in the next section.

3. The Proposed Approach

The discussion in the previous section suggests that a general purpose image database system should address several classes of retrieval needs, but all under a unified framework through a consistent user interface. The application areas should be able to choose only those functionalities that are both natural and useful to the domain much along the lines of EXODUS object-oriented database generator. An image, represented as an array of pixels (raster format) or as a collection of line segments (vector format), is considered to be at physical level representation. Since physical level representation provides very little information on the image contents without extensive image processing and understanding, it is desirable for an image database model to be able to provide multiple logical representations to facilitate interactive image retrieval. Logical representations denote abstractions of the image at various desired levels.

Most of the commercial systems operate at the physical representation level and build ad hoc logical representations for answering certain types of queries. The ad hoc logical representations vanish as soon as the query is processed and the whole process starts all over again when a similar query arrives next time. To avoid the exorbitant computational cost involved in building these ad hoc logical representations repeatedly some systems precompute and store important results which can be derived from such logical representations. However, it should be noted that this is not a solution since we cannot in general anticipate the precise nature of queries and even if we were to anticipate, simply it would be too voluminous and uneconomical to explicitly store all such precomputed data. Hence for practical and large spatial databases, multiple logical representations are not only necessary but also required to meet performance requirements for interactive processing

Each logical representation in this hierarchy can be viewed as an ideal representation for efficiently processing one or more classes of queries. Moreover, these logical representations can be very useful in restricting the access to the image data by assigning user access privileges to one or more layers in the hierarchy. In essence, this representation can also be regarded a view defining mechanism for image databases.

Now, we have, at one end of the spectrum, the physical layer at bit-level representation of the image. At the other end of the spectrum, we have the logical image which is an extremely abstracted version of the physical image. In between, we can conceive several logical layers corresponding to varying degrees of abstraction. The layers at lower levels embody more accurate representations of the image than the layers at the higher levels which provide a course representation by suppressing several insignificant details. By studying the application requirements and the limitations of the proposed approaches, we envision a multi-layered structure for retrieval. The various layers in the scheme are:

1. Physical Layer
2. Spatial and Shape Layer

3. Iconic and Attribute Layer
4. Conceptual Layer

These layers are not designed to operate in isolation but rather work in cooperation. To avoid redundancy in representation the layers are structured to form a lattice. The layers can also be viewed as multiple representations for the same object. We now illustrate the need for multiple logical representations by an example [42]. A street is considered as a spatial object for both city management and planning purposes. A traffic manager considers the city road network as an directed graph and he is only interested in flow analysis of the network and the spaces within the network (city blocks) are of no interest to him. The city cadastre department is interested in land parcels and their owners and perceives roads as boundaries for land parcels. Streets are viewed as surfaces by city personnel dealing with pavement revetments and the engineering department views them as 3D objects. We now briefly describe the layers and the class(es) of queries associated with them.

3.1 Physical Layer

The physical layer provides storage for the actual physical images. Each image is assigned an unique identifier for rapid retrieval. The association between an identifier and the corresponding pointer to the secondary storage where the physical image is actually stored is provided through a relational table. Usually no querying is permitted at this level. The relational table structure is created only to support the higher levels. Whenever a new image is added, the table is updated to reflect this addition.

3.2 Spatial and Shape Layer

This layer is the most complex in terms of the processing involved relative to the other layers. The types of queries that are supported by this layer are those based on shape similarity, those based on spatial similarity with respect to the spatial orientation of other objects, location based spatial queries, region based spatial queries, and an arbitrary combination of any of the above queries with non-spatial attributes and quantifiable spatial attributes. Another distinctive feature of this layer is the need for special techniques for query specification. For some query types such as location based and region based spatial queries, provision for query specification by directly operating on the analog form on the image is highly relevant. Iconic-based query technique which we describe in section four is suitable for queries for spatial similarity measurement. For queries which involve a combination of non-spatial attributes, quantifiable spatial attributes, and other spatial attributes, query specification schemes which combine operating directly on the analog form of the image method with SQL type of approach seems promising. All types of queries supported by this layer use spatial and topological reasoning in conjunction with the representational schemes available in the iconic and attribute layer.

3.3 Iconic and Attribute Layer

In some application areas involving overly complex and high resolution images and also in situations where the images are stored at a central location (server) but the server is required to provide access to the geographically distributed clients, the concept of an

iconic image is useful. An iconic image is an abstracted and symbolized version of the physical image. The iconic and attribute layer supports the following types of queries: browsing, queries based on non-spatial attributes such as the cost of a land parcel and queries based on quantifiable spatial attributes such as the volume of a lake. Browsing is implemented by having the image identifiers appear in one pane of a window in the form of a list box and the image (physical or iconic) corresponding to the user selection in the list box appear in another pane of the same window. If iconic images are used for browsing, the system displays the corresponding physical images at the user's request.

The relational table provided by the physical layer suffices to implement the browsing function. However, it should be noted that this type of browsing is not useful in applications where the entire database is just a single image extending in space as in geographic data applications. For such applications the browsing function provided by the spatial and shape layer is applicable.

All the objects in an image are referenced by an unique identification number and these identifiers are automatically assigned by the system. Each object is characterized by spatial attributes such as its spatial extent, minimum bounding rectangle, and centroid coordinates and by non-spatial attributes such as area, perimeter, and other attributes unique to an object. Spatial attributes may also include logical approximations, both general and domain specific, such as 2D-String approximation and medial axis representation. A tuple in a relational table is not suitable for storing object identification number and the associated spatial and non-spatial properties. This is because in an image database the number of different object types are very many and the number of instances per object type are very few. Moreover, the storage requirement even for the instances of the same object type differ depending upon the availability of attribute data and in some applications the attribute data is incrementally added. The above two aspects suggest storing both the spatial and non-spatial data as semi-structured text. Hence pattern text matching and reasoning with domain specific knowledge processes are required to extract the spatial and non-spatial attributes from the semi-structured attribute description. A representational scheme and the underlying storage access methods have to be designed in the light of these requirements.

The decision as to how much of spatial information is to be precomputed and stored as spatial attributes has to be carefully derived by considering the increased storage requirement versus the query processing time. Many a times the decision in favor of precomputing and explicitly storing only those spatial attributes which can be directly extracted from the geometrical description of the object alone is recommended. This approach explicitly eliminates the precomputing and storing of those spatial relationships among objects which can be obtained by other means such as spatial reasoning. This controlled approach to precomputation not only eliminates the computational effort involved in extracting the spatial attributes from the geometrical description but also obviates the need for searching the spatial index and the subsequent disk I/O involved in reading the geometrical information into the primary storage. Once the query is processed, the actual image can be displayed using the spatial index. Efficient and dynamic spatial indexing scheme is an integral part of the representation scheme for this layer.

3.4 Conceptual Layer

Conceptual layer is the most abstract layer and queries are specified using domain concepts. A restricted version of the natural language appears to be a suitable choice for query specification. Machine learning techniques are used to infer the higher level domain concepts from lower level primitive features and domain knowledge. Personal Construct theory has been successfully used by the authors in eliciting domain concepts from the domain expert [36].

In passing, it should be noted that this multi-layered architecture for retrieval is envisioned to be a general purpose model for use across several application domains. However, it is important to provide built-in extensibility into the model so that domain specific requirements can be easily accommodated.

4. Prototype System Implementation and Preliminary Results

To illustrate our scheme for retrieval based on spatial similarity, we selected forty-five residential floor plans as our image data. Each floor plan consists of various functional units (objects) such as bed rooms, living rooms, kitchen, and dining and esthetic units such as covered patio and porch. Each object is represented by the Cartesian coordinates of its centroid. The system is implemented using Smalltalk/V Windows which runs under a Graphical User Interface (GUI) known as Microsoft Windows. The system enables a user to interactively sketch an iconic query and then computes its spatial similarity with each and every image in the collection. The images in the collection are then shown in the decreasing order of their similarity for a specified query. An algorithm that was proposed in our previous work which runs in $O(n^2)$ time is used to compute the spatial similarity [35]; n being the number of objects in an image. The maximum possible similarity value of 100.00 indicates that the images are identical and smaller this value, lower is the similarity between the images.

The salient features of our system are:

1. A Browser to view images and the associated non-geometrical attributes
2. A Sketch Facility to interactively sketch or specify a query for spatial similarity retrieval
3. Provision for the display of images ranked by the similarity algorithm in a manner that facilitates user feedback to the system in the form of visual relevance judgments.

The browser window primarily consists of a list box to display the symbolic names of images in the database and a sketch area to display the selected image in the list box. When an image is displayed, the user can select an object in the image using a pointing device and be able to obtain non-geometrical attributes such as the floor area and ceiling height.

The purpose of the sketch facility is to enable the user to specify queries for spatial similarity. The system has an array of icons corresponding to the various functional and

esthetic units of residential dwellings (enlarged version of some icons are listed at the end of the references section). The user can select any of these icons and be able to place them anywhere in the sketch area of a window. Subsequent to this, non-geometric/non-spatial attributes can be assigned to these icons. In summary, by repeating this process the user can specify a spatial query by selecting any number of domain objects and spatially orienting them to suit his specification needs.

The results of the query are displayed by placing the query image in one pane of the display window and the database image in another pane of the same window. This configuration enables the user to visually compare the images and provide the system with feedback (this feature is not implemented at this time). Detailed discussion on this type of spatial query interface can be found in [45]. The user feedback has been successfully used in information retrieval systems to improve the quality of retrieval by using techniques such as query reformulation.

We found the system to be very easy to use and the ranking of the retrieved images quite well agree with our intuitive ranking. One application area where our system can be of immediate use is in real estate marketing. In huge metropolitan areas having large number of houses for sale, it is almost beyond the abilities of a human being to remember the spatial configuration of various functional and esthetic units in all the houses. The Realtors receive information on all the houses for sale through a service known as multiple listing service and this information does not contain any details on the floor plan design. Occasionally, some Realtors may be able to display from a video disk an image of the house taken from a vantage point. This only provides a general feeling for the quality of the neighborhood and the exterior of the house. However, it has been noted that some people prefer a house with a bedroom having orientation facing east so that waking up to the morning Sun is a psychologically pleasant experience. Yet some other people may prefer certain orientation for specific units in the house based on cultural and religious backgrounds. Our system can help the Realtors to quickly identify only those houses that closely match the preferences of the buyers.

5. Conclusions and Future Work

As of now we have implemented the spatial similarity aspect of the query language as well as parts of Iconic and Attribute Layer as it applies to spatial similarity computation. Implementation of the other aspects of the query language and formalizing the multi-layered layered architecture are the focus of our future research in this direction. Some aspects of the formalization include precisely defining associations between the query classes and layers, formal specification of the spatial query semantics, optimally structuring spatial data among layers, and controlling data redundancy through inheritance mechanism.

References

1. Berman, R. and Stonebraker, M. (1977), "GEO-QUEL: A System for the Manipulation and Display of Geographic Data," *Computer Graphics*, Vol. 11, No. 2, pp. 186-191.

2. Chamberlin, D. (1976), "SEQUEL-2 A Unified approach to data definition, manipulation and control," *IBM J. Res. & Dev.*, Vol. 20.No. 6, pp. 560-575.
3. Chang, N.S. and Fu, K.S. (1980), *Pictorial Information Systems*, Springer-Verlag, New York.
4. Chang, N.S. and Fu, K.S. (1980a) "Query by Pictorial Example," *IEEE Software*, Vol. 6, No. 6, pp. 519-524.
5. Chang, S.K. and Kunii, T. (1981) "Pictorial Database Systems," *IEEE Computer*, No.11, pp. 13-21.
6. Chang, N.S. and Fu, K.S. (1981), "Pictorial Query Languages for Pictorial Data-Base Systems," *IEEE Computer*, Vol. 14, No. 11, pp. 23-33.
7. Chang, N.S. (1981), *Image Analysis and Image Database Management*, UMI Research Press, Ann Arbor, MI.
8. Chang, S.K. (1989), *Principles of Pictorial Information Systems Design*, Prentice-Hall, Englewood Cliffs, NJ.
9. Chang, S.K., Jungert, E., and Li, Y. (1989) "The Design of Pictorial Database Based Upon the Theory of Symbolic Projections," *1989 Conference on Very Large Spatial Databases*, Springer-Verlag Publishing Company, Berlin.
10. Chock, M. (1982), *A Database Management System for Image Processing*, Ph.D. Dissertation, Department of Computer Science, University of California, Los Angeles.
11. Dilts, D. and Wu, W. (1991), "Using Knowledge-Based Technology to Integrate CIM Databases," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 3, No. 2, pp. 237-245.
12. Dueker, K. (1987), "Multipurpose Land Information Systems: Technical, Economic, and Institutional Issues," *Photogrammetric Engineering and Remote Sensing*, Vol. 53, No. 10, pp. 1361-1365.
13. Egenhofer, M. and Frank, A. (1988), "Towards a Spatial Query Language: User Interface Considerations," *14th VLDB Conference*, Los Angeles, CA., pp. 124-133.
14. Ehlers, M., Edwards, G. and Bedard, Y. (1989), "Integration of Remote Sensing with Geographic Information Systems: A Necessary Evolution," *Photogrammetric Engineering and Remote Sensing*, Vol. 55, No. 11, pp. 1619-1627.
15. Esposito, F., Malebra, D., and Semeraro, G. (1992), "Classification in Noisy Environments Using a Distance Measure Between Structural Symbolic Descriptions,"

- IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 3, pp. 390-402.
16. Felician, L. (1988), "Image Base Management System: A Promising Toll in the Large Office System Environment," *Data Base*, SIGBDP, Fall/Winter 1987/88, pp. 29-36.
 17. Grosky, W.I. and Mehrotra, R. (1989), Guest Editors, *IEEE Computer*, Vol. 22, No. 12, Special Issue on Image Database Management.
 18. Guenther, O. and Buchmann, A. (1990), Research Issues in Spatial Database, *SIGMOD RECORD*, Vol. 19, No. 4, pp. 61-68.
 19. Gunther, O. and Schek, H.-J. (1991), Advances in Spatial Databases, *Second Symposium on Spatial Databases*, Lecture Notes in Computer Science, Vol. 525, Springer-Verlag, New York.
 20. Ichikawa, T., Kikuno, T. and Hirakawa, M. (1980), "A Query Manipulation System for Image Data Retrieval by ARES," *IEEE Workshop on Picture Data Description and Management*, pp. 61-67.
 21. Iyengar, S.S. and Kashyap, R.L. (1988), Guest Editors, *IEEE Transactions on Software Engineering*, Special Issue on Image Databases.
 22. Joseph, T. and Cardenas, A. (1988), "PICQUERY: A High Level Query Language for Pictorial Database Management, *IEEE Transactions on Software Engineering*, Vol. 14, No. 5, pp. 639-650.
 23. Kasturi, R., Fernandez, R., Amlani, M. and Feng, W. (1989), "Map Data Processing in Geographic Information Systems," *IEEE Computer*, Vol. 22, No. 12, pp. 10-21.
 24. Kemper, A. and Wallrath, M. (1987), An Analysis of Geometric Modeling in Database Systems, *ACM Computing Surveys*, Vol. 19, No. 1, pp. 47-91.
 25. Lien, Y. and Harris, S. (1980), "Structured Implementation of an Image Query Language," *Pictorial Information Systems*, S.K. Chang and K.S. Fu (Ed.), Springer-Verlag, New York, pp. 416-430.
 26. Lindberg, D. (1979), *The Growth of Medical Information Systems in the United States*, Lexington Books, Lexington, MA.
 27. Mahoney, M. (1990), "Implementation of a Optical Disk System for Medical Record Storage," *Topics in Health Record Management*, Vol. 11, No. 1, Aspen Publishers, Frederick, Maryland.

28. McKeown, Jr. D (1988), "Some experiences Integrating Spatial Databases with Knowledge Based Systems," *GIS/LIS 1988*, pp. 765-773.
29. Meghini, C, Rabitti, F., and Thanos, C. (1991), "Conceptual Modeling of Multimedia Documents," *IEEE Computer*, Vol. 24, No. 10, pp. 23-29.
30. Menon, S. and Smith, T. (1989), "A Declarative Query Processor for Geographic Information Systems," *Photogrammetric Engineering and Remote Sensing*, Vol. 55, No. 11, pp. 1593-1600.
31. Motiwalla, J. and Narasimhalu, A.D. (1991) , *International Conference on Multimedia Information Systems*, McGraw-Hill, Singapore.
32. Nagy, G. and Wagle, S. (1979), "Geographic Data Processing," *Computing Surveys*, Vol. 11, No. 2, pp. 139-181.
33. Narasimhalu, A. and Christodoulakis, S. (1991), Guest Editors, *IEEE Computer*, Vol. 24, No. 10, Special Issue on Multimedia Information Systems.
34. O'Docherty, M. and Daskalakis, C. (1991), "Multimedia Information Systems: The Management and Semantic Retrieval of all Electronic Datatypes," *The Computer Journal*, Vol. 34, No. 3, pp. 225-238.
35. Raghavan, V.V. and Gudivada, V.N. (1990), "A Domain Independent Similarity Measure for Symbolic Images," *First Indian Computing Congress*, Hyderabad, India, November, pp. 195-203.
36. Raghavan, V.V., Gudivada, V.N., and Katiyar, A. (1991), "Discovery of Conceptual Categories in an Image Database," *International Conference on Intelligent Text and Image Handling*, RIAO 91, Barcelona, Spain, pp. 902-915.
37. Roussopoulos, N., Faloutsos, C. and Sellis, T. (1988), "An Efficient Pictorial Database System for PSQL," , *IEEE Transactions on Software Engineering*, Vol. 14, No. 5, pp. 630-638.
38. Stonebraker, M. and Rowe. L (1987) , "The POSTGRES Papers," Mem. No.UCB/ERL M86/85, University of California, Berkeley.
39. Stonebraker, M. & Dozier, J. (1991), "Large Capacity Object Servers to Support Global Change Research," Report #91/1, SEQUOIA 2000, Electronic Research Lab, University of California, Berkeley.
40. Stytz, M., Frieder, G. and Frieder, O. (1991), "Three-Dimensional Medical Imaging: Algorithms and Computer Systems," *ACM Computing Surveys*, Vol. 23, No. 4, pp. 421-499.

41. Tamura, H. and Yokoya, N. (1984), "Image Database Systems: A Survey," *Pattern Recognition*, Vol. 17, No. 1, pp. 29-43.
42. Thompson, D. (1991), "Geographic Information Systems: An Introduction," 1991-92 *International GIS SourcebookTM*, GIS World, Inc., Fort Collins, Colorado.
43. Tomlin, C. (1990), *Geographic Information Systems and Cartographic Modeling*, Prentice Hall, Englewood Cliffs, NJ.
44. Zloof, M. (1977), "Query-by-example: A Database Language," *IBM Systems Journal*, Vol. 16, No. 4, pp. 324-343.
45. Gudivada, V.N., Raghavan, V.V., and Carr, D. (1991), A Spatial Similarity Measure for Image Database Applications, Technical Report: 91-1, Department of Computer Science, Jackson State University, Jackson, MS.