Using 3D Spatial Relationships for Image Retrieval by Contents

M. Campanai, A. Del Bimbo, P. Nesi

Dipartimento di Sistemi e Informatica, Universita di Firenze
Via S. Marta 3 - 50139 Florence, Italy

Abstract

Iconic environments for querying pictorial data bases allow to express spatial relationships between the objects in the images through appropriate icon positioning. All the experiences reported in the literature use 2D icons and the query is directly mapped along with the symbolic image descriptions. With this approach, spatial relationships between the objects represented in the image could not be reconstructed exactly. Specifically, ambiguities arise for the retrieval of images of real world scenes with 3D objects. In this paper, a visual environment is presented for image retrieval from a database of real world pictures. The environment differs from the other experiences reported in the literature, since the visual query is performed taking into account relationships between the objects in the scene that is represented in the image, rather than between the image subparts. A 3D iconic interface with 3D icons allows the user to define a virtual scene, which is used to perform the query. Images are stored in the database along with the descriptions of the 3D scenes they represent.

1 Introduction

Retrieval of images from *pictorial databases* can be performed in a natural way on the basis of their contents. Basic design issues are related to the way in which pictures are represented in the database [1], [2], [3], [4] and how the query is performed. Examples of query processing based on content in image database applications are reported, among the others, in [3], [4], [5]. In [3], and [4], the selection of images is performed through spatial predicates that are directly mapped with the image data. A different approach is proposed in Chang [5], where each image is associated with a data structure, a *2D string* output of a 2D spatial analyzer, which preserves the embedded spatial knowledge. 2D strings capture the ordering relationships between image subparts along the X- and Y-axis [1], [2]. Picture queries can also be specified in terms of 2D strings. A visual environment with 2D icons is used to define the spatial relationships between objects in the requested images. The 2D string associated with the iconic reconstruction is matched with the 2D strings associated with images. The same approach is further developed in [6], where uncertainty management is introduced using fuzzy-set techniques.

Picture representation through symbolic descriptions in the Chang's style, provides an effective and natural way to construct iconic indexes for pictures. Moreover, the proposed approach combines the benefits of querying-by-example and of icon-oriented visual programming [7].

However, while the use of 2D icons to reproduce images is effective for images representing 2D objects or very thin 3D objects, this could not allow to exactly define spatial relationships in the image for the case of images representing real world scenes with 3D objects.

In this paper, a new 3D iconic environment for the retrieval of images of real world scenes from an object-oriented pictorial database, is presented. The environment differs from the other experiences reported in the literature, since the visual query is performed taking into consideration the spatial relationships between the objects in the scene that is represented in the image, rather than between the image subparts. Retrieval is performed by placing 3D icons into a 3D virtual space to reproduce the original real scene. The symbolic description of the virtual scene is hence compared along with those associated with the images in the database. A visual language and its compiler have been designed and implemented which support the definition and interpretation of the visual query, respectively.

The system is designed according to the object-oriented paradigm and is implemented in C++ on a

1049-2615/92 $3.00

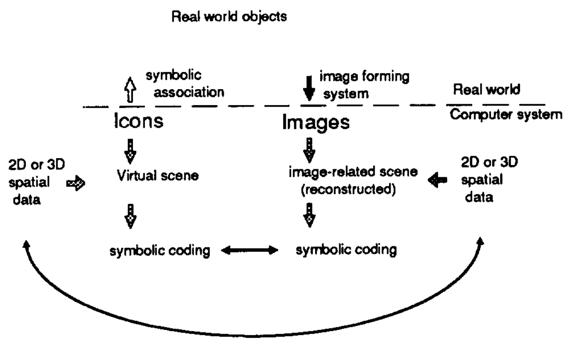


Figure 1: Image querying through icons: the dimensionality of the icons must follow that of the objects in the scene represented in the image.

PC/AT platform. It is interfaced with the Ontos Data Base Management System, running on a separate system in the same LAN. The virtual world visualization is obtained by using the Matrox PG-SM-1281 board. The pointer is a custom-modified manipulation glove.

2 Using 2D or 3D Icons

A basic point in image database querying is that of understanding what spatial relationships should be associated with images.

As a general statement, differently from what is currently assumed in the literature, we affirm that an unambiguous correspondence is established between icons and image contents, provided that images are described in terms of the original scene rather than of their subparts, and combinations of icons represent virtual scenes rather than virtual images. This leads to state that the dimensionality of data structures associated with icons must follow the dimensionality of the objects in the scene represented in the image (Fig.1).

For the case of images with 2D scene data, an exact correspondence exists between spatial positions of icons and positions of objects in the image, when 2D icons are used. Therefore, a 2D iconic environment is able to express the query unambiguously.

In this same case, the use of 3D icons is not justified by the nature of the problem.

For the case of images with 3D scene data, a difference exists as to whether image or scene data are considered. Relationships between image subparts are not the same as those between scene objects, the former being evaluated on the 2D image plane and the latter in the 3D scene space. Generally speaking, a single combination of 2D icons may correspond to several displacements of 3D spatial data, due to the impossibility of reproducing the scene depth using 2D icons. According to this, queries could not be expressed in an exact way and several images in the database could exist, corresponding to the scene defined in the virtual plane. These drawbacks could be somehow overcome using icon overlapping to reproduce the missing third dimension. Unfortunately, overlapping can be used only for simple scenes since it impacts on the query intelligibility.

If 3D icons are used, a 3D querying environment is required. 3D icons are placed in the virtual space to reconstruct the scene represented in the image. Spatial relationships between 3D icons exactly represent spatial relationships between the objects in the scene. Therefore, once the point of view has been set, the image representing such a scene can be uniquely identified, provided that the scene description has been associated with the image.

It should be noted that using 3D icons and their spatial relationships complies with the typical behavior of the user, who expresses the query on the basis of the view of the scene he has in his mind, which is 3D. Therefore, in this approach, the user is relieved from the burden of performing a translation of his scene view into a 2D view, such as that represented in the image.

3 Symbolic Coding of 3D Spatial Relationships

An important point in the definition of a spatial syntax is object representation. Different approaches to object representation have been proposed. In [2], [5], and [8], 2D spatial structures within the image were projected along each coordinate axis. Specifically, in [2], the centroids of the objects were employed so that their spatial extent was reduced to a point. In [5] and [8], minimum enclosing rectangles were used to represent spatial structures in the image. In our approach, 3D objects in the original scene and 3D icons are represented by the projections along the three coordinate axes of their minimum enclosing parallelepipeds (MEPs). Their spatial relationships are

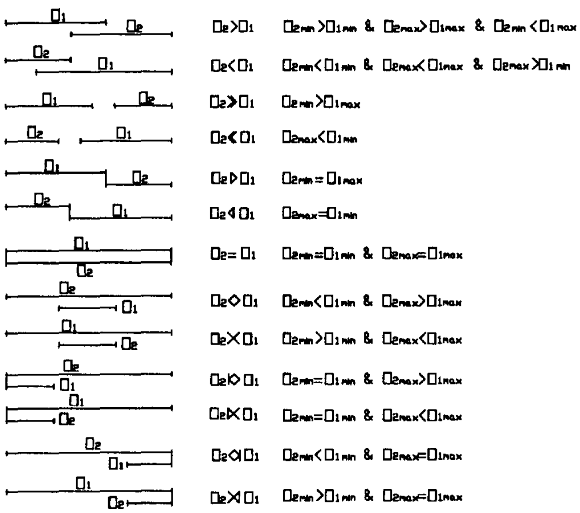


Figure 2: Binary spatial ordering relationships between objects. For the sake of simplicity projections of MEPs are depicted with reference to a single axis.

encoded according to a new 3D syntax. A 2D syntax can be immediately derived from this in order to represent 2D scenes.

3.1 Relationships between objects

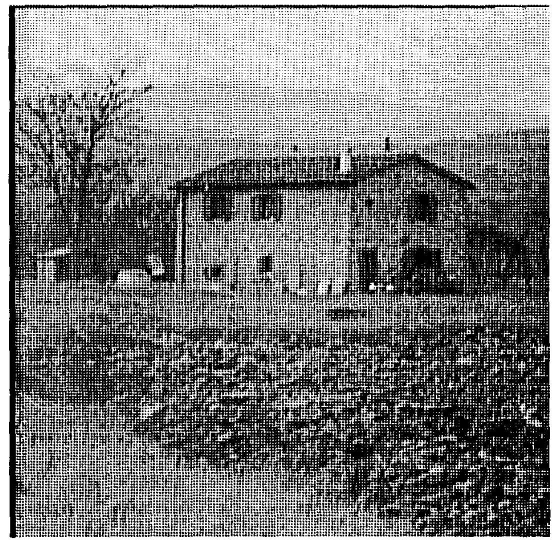
For each couple of objects O_1, O_2, in the imaged scene a 3D spatial relationship is defined through a quadruple R, evaluated according to Cartesian coordinate system with the X-and Y-axis aligned with the corresponding axes of the the camera coordinate system and the Z-axis tilted by a θ angle with respect to the camera optical axis in order to lay on the scene plane.

$R = (OP_{S_x}, OP_{S_y}, OP_{S_z}, OP_T)$
where :

- OP_{S_k} is any symbol taken from S = { ≫ , ≪ , > , < , ⊳ , ⊲ , = , ◇ , × , |◇ , ◇| , ⋊ , ⋉ }, which defines a spatial ordering relationship between two objects along the k-direction of the reference Cartesian system of coordinates, as shown in Fig.2, where, for the sake of simplicity, reference is made only to the MEP projections along the X-axis:
- OP_T is any symbol taken from T = { ||, | }, which defines a global spatial relationship between two objects, being:

  || the symbol of parallelism

  | the symbol of collinearity



tree - house: (≪ , |◇ , ≪ ,)
house - car : (◇ , |◇ , ◇ ,)
tree - car : (≪ , |◇ , ≪ ,)

Figure 3: Sample image representing a real world scene with 3D objects and its description.

Parallelism and collinearity relationships between couples of objects are defined considering parallelism and collinearity of the planes passing through the centroid of the MEP and parallel to the MEP axis in the maximum elongation direction.

A sample image is shown in Fig.3 with the corresponding description of the 3D imaged scene It should be noticed that this description is exactly the same as that which is perceived by an observer when viewing the same scene in the real world from the same viewpoint.

4 Image Retrieval Through 3D Icons

In the following, a 3D iconic environment for querying a database of images of real world scenes is presented.

Images and their descriptions are stored into an object-oriented database. Descriptions can be obtained automatically through a scene understanding task, by using multiple images. However, the scene

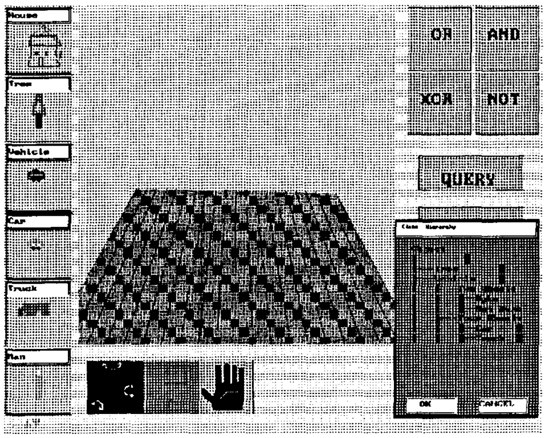


Figure 4: The system user interface. From left to right, IES, FS, CI, DC, LOS windows of the Dialogue_Interface. The Query_Window is in the central part of the screen.

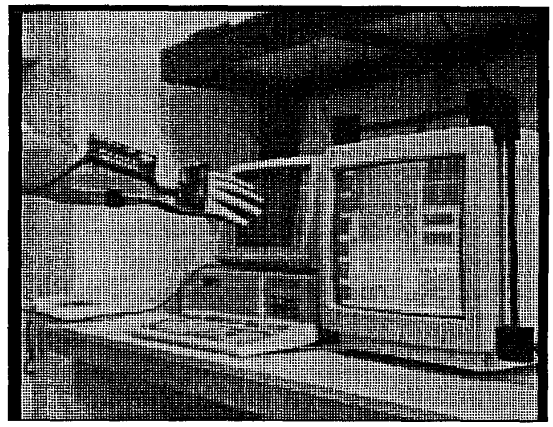


Figure 5: Interaction with the system using the manipulating glove.

understanding system for the derivation of scene relationships has not been linked to the present version of our system. Therefore, the scene descriptions have been defined manually.

Icons are stored in the database as well. They represent categories of objects (category_icons) or object instances (object_icons). Icons are comprised of a graphical form and a text body [7], the latter specifying the attributes of the entity represented by the icon. Icon classes are arranged into an inheritance hierarchy. The inheritance hierarchy can be inspected during the icon selection step of the query construction or for the definition of new icon classes.

The visual interface of the system is comprised of a *Dialogue_Interface*, which supports the selection of icons from the database and several operating facilities, as well as of a *Query_Window*, which is used for the definition and the visualization of the 3D query, as shown in Fig.4. To add realism to the user's interaction, a *manipulating glove* interface was introduced to perform visual database queries

When operations are made in the Query_Window, a 3D cursor follows the movements of the human operator hand and allows rotation, translation and grasping operations. An overview of the visual language for these operations is given in the Appendix with a modified BNF notation.When used in the Dialogue_Interface, the glove simply operates as a 2D pointing device and an arrow cursor is visualized.

4.1 Operations in the Dialogue_Interface

The Dialogue_Interface section is organized into several distinct windows (Fig.4):

- an*Iconic_Entity-type_Selection (IES)* window (in the left part of the screen) including 3D icons organized in a list, to be selected for the definition of the query.
- Three *Facilities_Selection (FS)* windows (in the lower part of the screen), for the change of the point of view, the zooming of the scene, and the rotation of the hand in the virtual space.
- a *Class-hierarchy_Inspection (CI)* window (in the lower right part of the screen), where the user can search for the desired icon inspecting the database icon class-hierarchy.
- a *Database_Command (DC)* window, in which two buttons for the issue of commands for the execution of the query and the display of the icon class-hierarchy are present.

- a *Logic_Operation_Selection (LOS)* window (in the upper right part of the screen), including icons of logical operators AND, OR, XOR, NOT to be used to define complex visual queries.

4.2 Operations in the Query_window

Queries on the pictorial database are expressed by example. The user can pose the selected icon in any place of the virtual space to reconstruct the real scene represented in the images to be extracted from the database. Spatial operators between two icons such as *Right, Behind, In_front_of, Above* .... are naturally defined by the user locating the 3D icons at the appropriate positions in the space. When the icon of interest is selected, the text body of the corresponding class is presented to the user for a possible specification of logical or numerical constraints on the attributes values, in order to restrict the query scope (see Fig.6). In this case, the cursor is 2D and the keyboard is enabled. Only images in which objects satisfy both spatial and logical or numerical constraints are answered. An example of visual query is reported in Figs.7-8. In Fig.7, several 3D icons corresponding to the objects of interest are placed in the virtual space in order to specify a query such as: ***" search all the images which include a house, a tree behind on the right-hand-side, and a vehicle in the lee of the house on the left-hand-side"***.To this end, the icon of a house has been selected through the manipulation glove and located on the ground icon. Then, the tree icon has been selected and placed behind on the right-hand-side and the vehicle icon has been placed on the left-hand-side in the lee of the house. Icons positions are evaluated with reference to a fixed Cartesian coordinate system provided by the graphic processor. The user can change his point of observation of the scene (i.e. the position of a virtual camera) during the query construction stage, using the point-of-view-change facility of the system. Once the position and orientation of the camera are chosen, spatial relationships between couples of icons are evaluated considering MEP projections with reference to a Cartesian coordinate system with the X-and Y-axis aligned with the corresponding axes of the the camera coordinate system and the Z-axis tilted by a θ angle with respect to the camera optical axis in order to lay on the plane of the virtual scene. The image of the required scene is presented on the screen after the query parsing.

It follows that the user can issue a query for an image of the same scene from a different point of view, simply using the point-of-view-change facility, and thus avoiding the rewriting of the query. In Fig.9,

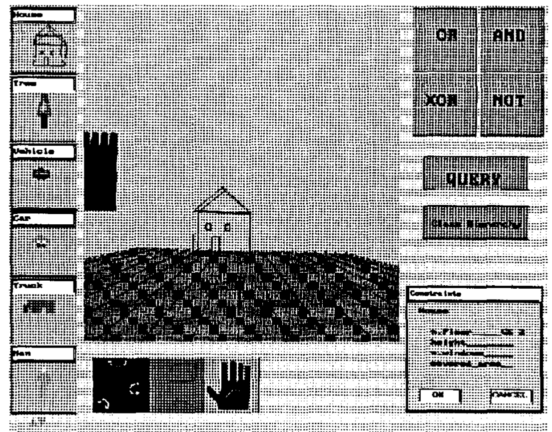


Figure 6: The selection of an icon makes the icon text body to be presented to the user for the specification of attribute values or constraints.

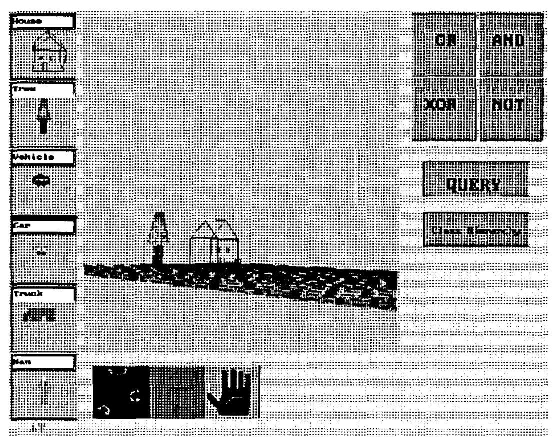


Figure 7: Definition of a virtual scene by placing selected icons in the virtual space of the Query_Window.

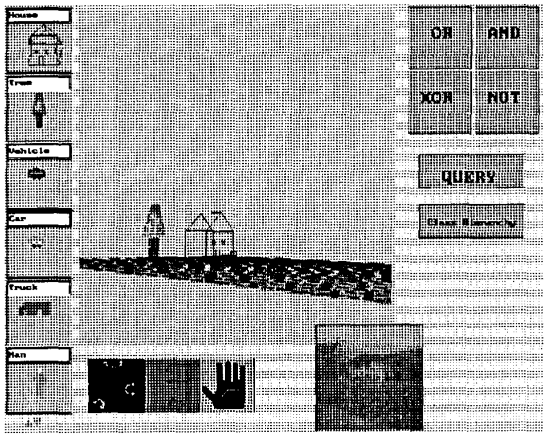


Figure 8: Query results.

a different query is issued, by rotating the previous virtual scene. From the new viewpoint, the vehicle icon is hidden by the house icon, and therefore the new query translates in: *"select all the images which include a house with a tree in front of the house on the left-hand-side"* . The query answer (Fig.10) presents two different images.

Please note that in a single visual query, AND, NOT logical operators are naturally defined through the presence and absence of icons in the virtual scene, respectively. The logical operator AND, OR, XOR, NOT can be used to form a more complex query from two separate queries.

5 Conclusions

A visual environment for the retrieval of images of 3D real world scenes from an object-oriented database has been presented. A picture query is also specified as a symbolic sentence. As a different approach from what is commonly assumed in the literature, symbolic descriptions associated to the images refer to spatial relationships in the scene that is represented in the image, rather than to the image subparts. With this approach, retrieval is performed without any ambiguity. Therefore, the visual environment is made up a 3D virtual scene and 3D icons are used to express the query. The user can place icons in the virtual space

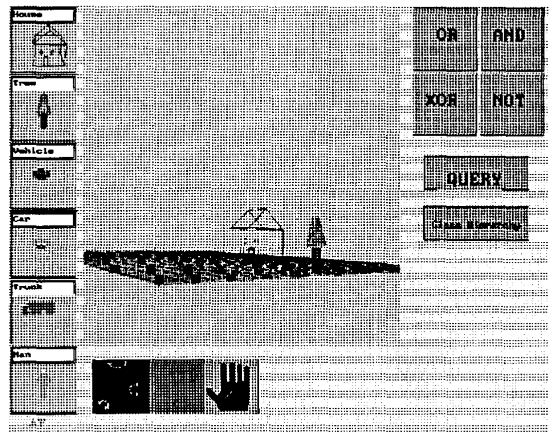


Figure 9: A new query can be issued simply changing the point of view of the scene previously defined.

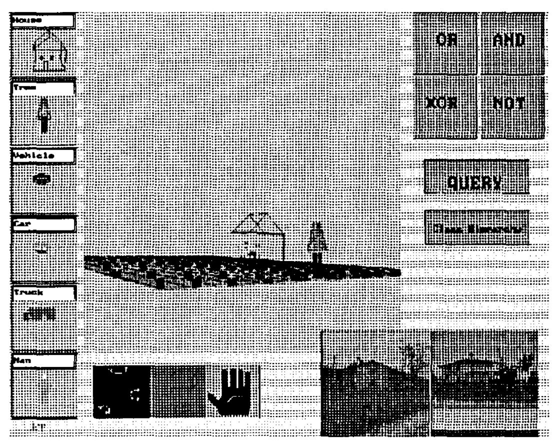


Figure 10: Query results.

to reproduce the spatial relationships between the objects in the imaged scene. This also complies with the common behavior of the user who expresses the query on the basis of the view of the scene he has in his mind, which is actually 3D.

A Appendix

A visual language and its compiler were designed and implemented which support database querying. The definition of the language for the operations in the Query_Window is reported in the following through a modified BNF notation:

```
<query_expression> ::= <single_query> | <complex_query>
<complex_query> ::=
        <single_query><boolean><single_query> |
        <complex_query><boolean><single_query>
<single_query> ::= [NOT]
        ( { <icon><icon_unary_spatial_op> } |
        { <icon><icon_binary_spatial_op><icon> }) |
        <list_icon><list_unary_op> |
        <object_icon><object_icon_op>
<boolean> ::= AND | OR | XOR
<icon> ::= <object_icon> | <class_icon> [<class_icon_op>]
<icon_unary_spatial_op> ::= <icon_position> | everywhere
<icon_position> :: = xyz-MEP projections,
                     xyz-MEP centroid coordinates,
                     max elongation axis angles
<icon_binary_spatial_op> ::= {<S_spatial_op> }^3
                             <T_spatial_op>
<S_spatial_op> ::= ≫ | ≪ | > | < | ▷ | ◁ | = |
                   ◇ | ╳ | |◇ | ◇| | ╳| | |╳
<T_spatial_op> ::= ‖ | |
<class_icon_op> ::= {<attribute><operator><value>}
<attribute> ::= ATTRIBUTE_FROM (<class>)
<operator> ::= <numeric_operator> | <symbolic_operator>
<numeric_operator> ::= = | ≠ | > | < | >= | <=
<symbolic_operator> ::= EQ | NE | GT |
                        LT | GE | LE
<value> ::= constant
<list_unary_op> ::= SHOW_LIST | EXTRACT |
                    <list_constraint>
<list_constraint> ::= {<single_query>}
<object_icon_op> ::= SHOW_CLASS_DESCRIPTION
```

References

[1] S.-K. Chang and S.-H. Liu, "Picture indexing and abstraction techniques for pictorial database," *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 6, pp. 475–484, July 1984.

[2] S.-K. Chang and C. Yan, "Iconic indexing by 2-d strings," *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 9, pp. 413–428, May 1987.

[3] J. A. Orenstein and F. A. Manola, "Probe spatial data modeling and query processing in an image database application," *IEEE Transactions on Software Engineering*, vol. 14, pp. 611–629, May 1988.

[4] N. Roussopoulos, C. Faloutsos, and T. Sellis, "An efficient pictorial database system for psql," *IEEE Transactions on Software Engineering*, vol. 14, pp. 639–650, May 1988.

[5] S.-K. Chang, C. W. Yan, D. C. Dimitroff, and T. Arndt, "An intelligent image database system," *IEEE Transactions on Software Engineering*, vol. 14, pp. 681–688, May 1988.

[6] S.-Y. Lee, M.-K. Shan, and W.-P. Pang, "Similarity retrieval of iconic image database," *Pattern Recognition*, vol. 22, no. 6, pp. 675–682, 1989.

[7] S. K. Chang, "Principles of visual languages," in *Visual Programming Systems* (S. K. Chang, ed.), pp. 1–59, NJ, USA: Prentice Hall Ed., Englewood Cliffs, 1990.

[8] E. Jungert, "Extended symbolic projections as a knowledge structure for spatial reasoning and planning," in *Pattern Recognition* (J. Kittler, ed.), New York, NY, USA: Springer Verlag, 1988.