Relation-Based Similarity

Dimitris Papadias
Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
dimitris@cs.ust.hk

Vasilis Delis
Computer Engineering and Informatics Dept.
University of Patras
and Computer Technology Institute
Kolokotroni 3, 26221, Patras, Greece
delis@cti.gr

Abstract: Similarity queries constitute an active area in spatial query processing. The paper addresses the problem of qualitative similarity based on spatial relations. On the one hand spatial similarity entails mechanisms for representing and reasoning on spatial relations, while on the other it introduces a high level of uncertainty. Several spatial queries can be rather fuzzy and user or application dependent. Moreover, relations such as *near*, *northeast* etc. lack universally accepted semantics, and as a result their processing in Spatial Databases and GIS has to provide a high level of flexibility in order to satisfy real-life needs. Our work extends the notion of conceptual neighbourhood (originally defined for 1D space) to include higher dimensions and proposes a unified multiresolution framework for the handling of topological, directional and distance relations. We discuss how object and image similarity queries can be effectively handled and show how uncertainty can be seamlessly incorporated in our model.

1. INTRODUCTION

This paper studies similarity retrieval based on spatial relations. We assume a database of images (or maps) of distinct objects with certain properties, and queries that retrieve objects and images that satisfy some either well-defined or fuzzy characteristics, the latter being more interesting and usually more difficult to handle: e.g., "find all industrial zones *near* a lake in area A" or "find all images *similar* to image I". The expected output of both queries is not well-defined, but depends on the particular user's notion of *near* and *similar*. This work uses relation similarity in order to model this type of uncertainty. Relation-based similarity is qualitative, because it does not rely on absolute coordinates and actual shapes, but on relative objects' positions.

The spatial relations that we assume in this work are *projection-based* extensions (Papadias and Egenhofer, 1997) of Allen's (1983) work on (convex and closed) temporal intervals. Freksa (1992) defined the conceptual neighbourhood for Allen's 13 *primitive* relations as shown in Figure 1. If we start with relation R_1 and extend (or move) the upper interval to the right we derive relation R_2. With a similar extension we can derive the transition from R_2 to R_3 and so on. R_1 and R_3 are called 1st degree neighbours of R_2. Previous work on spatial relation similarity has been carried out for topological relations (Egenhofer and Al-Taha, 1992) and for classes of both topological and direction relations (Bruns and Egenhofer, 1996).

In Proceedings of the 5th ACM Workshop on GIS, Las Vegas, ACM Press, 1997.

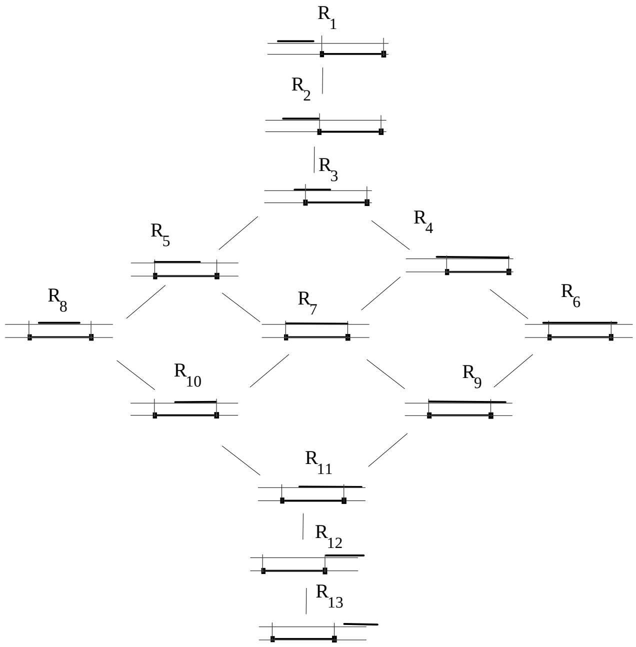

Figure 1 Conceptual neighbourhood in 1D space

Our approach is novel in that it covers all three types of spatial relations (topological, directional, distance) and unlike previous methods that deal mostly with modelling, shows how relation similarity can be used in practical query processing involving object and image retrieval. The rest of the paper is organized as follows: Section 2 extends conceptual neighbourhood to higher dimensions and various granularity levels. Section 3 describes object retrieval, and section 4 image retrieval using relation similarity. Finally, Section 5 concludes the paper.

2. EXTENSIONS OF NEIGHBOURHOODS

The extension of 1D relations to N dimensions is straightforward. A *N*-dimensional relation is an N-tuple of 1D relations, e.g. for a 2D relation: $R_{2\text{-}3} = (R_2,R_3)$. Each of the 1D relations in the tuple corresponds to the spatial relationship between the *N*-dimensional objects in one of the dimensions. So if *s* is the number of possible 1D relations at a particular resolution, the number of *N*-D *primitive* relations that can be defined at the same resolution is s^N.

The neighbours of a *N*-dimensional relation, consist of the set of 1D neighbours in each dimension. For example, the neighbours of $R_{1\text{-}4}$ are $R_{1\text{-}3}$, $R_{1\text{-}6}$, $R_{1\text{-}7}$ and $R_{2\text{-}4}$. Since in one dimension a relation may have as many as four neighbours, in N dimensions every relation has a maximum of 4*N neighbouring relations.

The above concept of neighbourhood is limited in the sense that it cannot deal with distances. If, for example, two objects are disjoint in both projections, their distance is not important at determining their relative relation. In order to overcome this deficiency we may incorporate additional relations in the framework. Figure 2 shows how we can replace R_1 with 5 new relations. R_{1_1} implies that the upper (primary) interval is before the reference interval but totally included within δ distance from the left point. This relation could be interpreted as *near* (δ is application dependent). R_{1_5} means that all points of primary interval are outside δ (this can be interpreted as *far*). The intermediate relations can have similar interpretations. Figure 2b illustrates the application of the same idea to 2D space, where a grid of side δ is formed around the reference object.

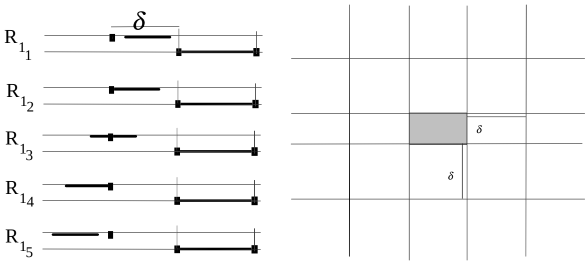

Figure 2 Example of distance-enhanced neighbourhoods

Figure 3 illustrates the complete neighbourhood graph involving distance extensions (each original relation is replaced by its refinements). We can use an arbitrary number of, possibly different, interval-extensions (δ) to define as many relations as needed to match the application needs. According to the requirements of the particular application, not all dimensions might be tuned at the same resolution, in which case the maximum number of *N*-D relations is the product of the corresponding numbers for each dimension.

The *degree of neighbourhood* (or conceptual distance) $d(R_i,R_j)$ between any two primitive relations R_i and R_j is equal to their distance in the conceptual neighbourhood graph. A *relation set r*, represents a disjunction of relations. The distance between a relation set r and a primitive relation R is the minimum distance between any relation of the relation set and R, i.e. $d(r,R) = \min_{R_k \in r}(d(R_k, R))$. Each real-world spatial relationship is mapped onto a relation set, which is defined in advance and is the same for all users (an agreement must be made regarding which relations must be associated to a particular spatial relationship - e.g. *northeast* could be potentially assigned the relation set $\{R_{1\text{-}13}\}$, or, $\{R_{1\text{-}13}, R_{2\text{-}13}, R_{1\text{-}12}\}$, etc.). It constitutes a kind of core for the corresponding spatial relationship in the sense that all the primitive relations of the set satisfy the spatial relationship.

Projection-based definitions of similarity are particularly suitable for spatial database retrieval because very often in practical applications, minimum bounding rectangles (MBRs) are used to approximate actual objects. Because MBRs are projection-based approximations, our methods are directly applicable for this type of query processing. In the next sections we apply the above concepts in order to answer two types of queries: relation-based object retrieval and image similarity retrieval.

3. RELATION-BASED OBJECT RETRIEVAL

An important class of queries in Spatial Databases and GIS consists of queries of the form: "find all (*primary*) objects that satisfy a given relation set *r* with respect to a given *reference* object *X*". The problem with such queries is that usually spatial relations cannot be modelled by boolean domains, but the difference between objects that satisfy the query, and the ones that don't, may be quantifiable and gradual. Information retrieval techniques (like WWW search engines), deal with this problem by associating the retrieved documents with a score proportional to the similarity of the query and the document (Salton et al., 1994). In an analogous manner, the output objects in the case of spatial queries should have an associated "score" to indicate the similarity between their relation and the target relations of the query relation set. Here the score is inversely proportional to the degree of neighbourhood (distance in the graph).

To illustrate the above ideas, we consider a situation of a flooded city where the mass of water moves abruptly and a GIS for damage assessment. The database contains images (Figure 4) describing instances of the flood at different times, and the queries are of the form " find all residential areas *covered by* the flood at any time (in any image)". Figure 4 also includes a simple object-class hierarchy. Objects in queries can be specific *instances* (e.g., flood) or *generic classes* (residential area) or disjunctions thereof (residential area or commercial center).

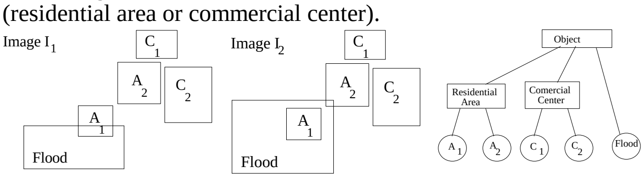

Figure 4 Example images

There are two types of uncertainty associated with object retrieval. The first one is due to the inherent fuzziness of some spatial relations. Assume that *Northeast,* is mapped onto the relation set $\{R_{1\text{-}13}\}$; all objects that are related with $R_{1\text{-}13}$ with respect to a reference object *X* are said to be *northeast* of *X*. However, additional neighbouring relations (such as $R_{1\text{-}12}$, $R_{2\text{-}13}$) may also be regarded as *northeast,* depending on the shape and the size of the object, or the user's expectations. The corresponding objects should be retrieved (with a lower score) and the user should decide about their relevance to the query.

The second type of uncertainty is due to fuzzy boundaries. Several spatial objects do not have well defined boundaries (e.g., residential areas, forests). Others have boundaries that change over time (shorelines in the presence of tide). In such cases the stored objects are only approximations of actual ones (they can be

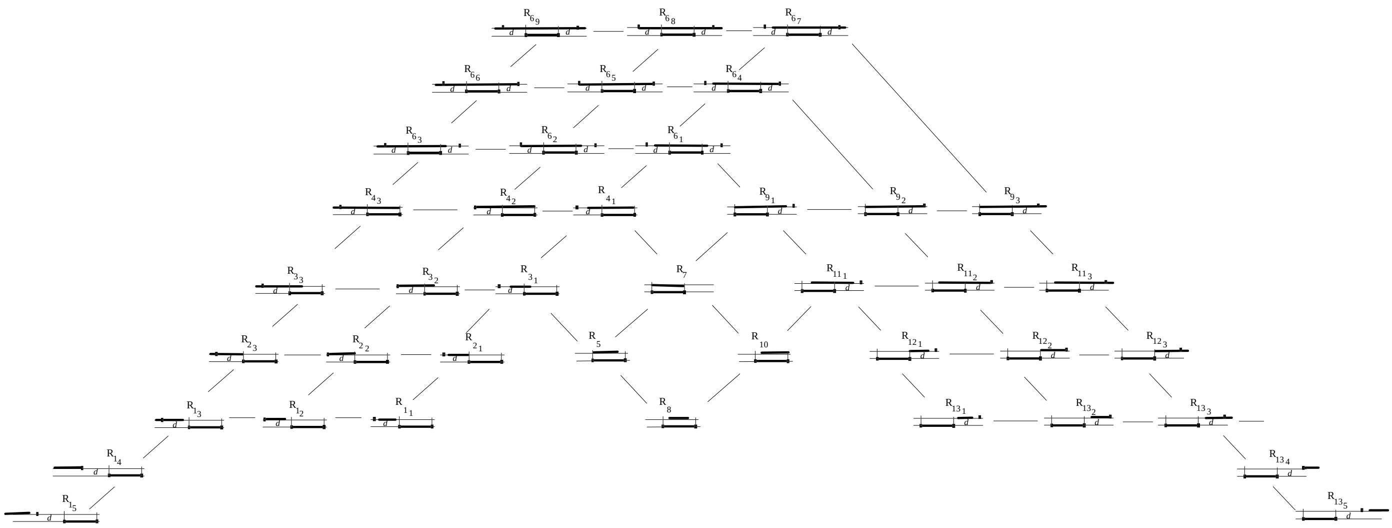

Figure 3 Neighbourhoods with distances

slightly larger or smaller), and as a consequence, the derived projection relations may be inaccurate. Conceptual neighbourhoods can deal with both forms of uncertainty.

Formally an object retrieval query Q_O is a 5-tuple (C_Y, *r*, *X*, *i*, *degree*). The result of the query consists of the retrieved object instances which belong to the class C_Y and satisfy the relation set *r* with respect to the object class or instance *X*, in a set of images *i*, at a maximum tolerance (relation distance) *degree*. Let *x* be a retrieved object and X_I an instantiation of *X* (unless *X* is already an instance in which case $X_I = X$). The score of *x* is MAX_Degree_of_Neighbourhood (which is the maximum distance in the neighbourhood graph and equals to *8* in 1D and *16* in 2D assuming Allen's relations) minus the minimum distance between $R(X_I, x)$ and the relation set *r*. The following pseudo-code implements the above described object retrieval. *Find_Instantiations* function returns the domain of possible instantiations of the reference object in an image.

```
Object_Retrieval(Generic_Primary_Object C_Y, Relation_Set r,
 Reference_Object X, Image_Set i,real Degree){
Object_Set result={};
for each object Y: score(Y)=0;
for each image I in i {
  Find_Instantiations(I,X);
  for each instance A in Domain(I,X){
  for each primitive relation R_kl in r
    used(R_kl)=FALSE; counter=0;
    do {
      for each primitive relation R_kl in r such that not(used(R_kl)){
        for each object Y such that R_kl(Y,A) in I and C_Y=superclass(Y){
          current_score= MAX_ Degree_of_Neighbourhood - counter;
          if (current_score>score(Y)) score(Y) = current_score;
                  result = result ∪ Y;}
         r=r ∪ Neighbours(R_kl);
         used(R_kl)= TRUE;
          }
      counter++;
      } while (counter<= Degree);
    } // end-for reference objects
} // end-for images
return result;

Find_Instantiations(Image I, Object X){
if (X is object class) {
              let C_x be the subclass(es) of X;
              Domain(I,X)={A / A ∈ I and A ∈ C_x};
} else { // X is instance//
         if (X ∈ I) Domain(I,X)={X};
         else Domain(I,X)= {X∅};// empty Instantiation. }
return Domain(I,X) // returns the possible instantiations of X in image I
}
```

Figure 5 illustrates the processing of the query "Find all residential areas or commercial centers *northeast* of the flood in any image with a target degree 2" (`Object_Retrieval (Residential Areas∨Commercial Centers, {R1-13}, Flood , {I1, I2}, 2)`). Initially image I_1 is processed and Domain(I_1,Flood)={Flood}, *r*={R_{1-13}}. C_1 is retrieved because the relation R_{1-13}(C_1,Flood) holds, with a score equal to MAX_Degree_of_Neighbourhood (16). The neighbours of R_{1-13} (R_{1-12}, R_{2-13}) are then added to *r*, and R_{1-13} becomes *used*, in order to avoid retrieving object C_1 again (used relations are denoted with italics in Figure 5). During the second iteration of the *do-loop* (*counter*=1) C_2 is retrieved, because R_{2-13} (C_2,Flood). The last object to be retrieved from image I_1 is A_2 with a score of 14. Then I_2 is processed and the same procedure is followed.

	Relation Set	Primary Object	Relation	Score	Current Score
I_1	R_{1-13}	C_1	R_{1-13}	16	16
I_1	*R_{1-13}*, R_{1-12}, R_{2-13}	C_2	R_{2-13}	15	15
I_1	*R_{1-13}*, *R_{1-12}*, *R_{2-13}*, R_{1-11}, R_{3-13}	A_2	R_{1-11}	14	14
I_2	R_{1-13}	C_1	R_{1-13}	16	16
I_2	*R_{1-13}*, R_{1-12}, R_{2-13}	-	-	-	-
I_2	*R_{1-13}*, *R_{1-12}*, *R_{2-13}*, R_{1-11}, R_{3-13}	C_2	R_{2-13}	15	14

Figure 5 Example query processing

The above example does not represent the most general *Object_Retrieval* query, since it involves only one (instance) reference object. In case of a reference object class, we would have to perform the same processing for each instantiation of the reference object, in every image. The cost *C* of retrieval of all objects that satisfy a relation set in an image, depends on the relation and the underlying data structure. This cost is measured by the number of disk accesses and, in general, increases with the area covered by the projection relation. The previous algorithm calls the spatial data structure a number of times equal to *m = Number_of_Images * Number_of_Instantiations_in _Each_Image * Degree*, resulting in a total cost of *m*C.*

Unlike the fuzziness of direction relations, for topological relations there exists a set of formal and widely accepted definitions based on the *intersection* model (Egenhofer and Franzosa, 1991). Although, assuming the intersection model, the expected output for topological queries is unambiguous, conceptual neighbourhoods are still useful to deal with fuzzy boundaries. Consider the query: "find all residential *covered by* the flood in any image". The MBRs that contain potential answers to the query (in the ideal situation) are the ones *covered_by*, *inside*, or *equal* to the MBR of flood. In the presence of fuzzy boundaries, however, we need to retrieve some additional MBRs that satisfy neighbouring relations, in order to make sure that we don't miss any objects (for details see Papadias et al., 1995). All queries involving combinations of spatial relations with respect to a reference object can be derived by proper application of *Object_Retrieval*. In case of disjunctions of spatial relationships, the relation set consists of the union of the relation sets for each relationship. For conjunctions (e.g., *northeast* and *covered by*), the input of the primitive query is the intersection of the individual sets.

4. RELATION-BASED IMAGE RETRIEVAL

Another important class of queries retrieves images that satisfy a set of given relations between distinct objects. e.g., "*find all images* where some residential area *southwest* of a commercial center is *covered_by* the flood". Our interest is not to give a binary answer (yes, no) when assessing an image. Rather, a more interesting (and useful in practice) attempt would be to rank all images according to their resemblance to the queried spatial configuration. Under this perspective, the above query is essentially an image similarity query, since it describes a generic *query-image* (depicting some residential area southwest of a commercial center, covered by the flood) which is matched with the stored images in order to retrieve the most similar ones. Similarity is only based on the properties of the objects to be matched (e.g., residential area,…) and their interrelationships but not on their visual characteristics (e.g. shape, size).

Such queries can be formalised as finite sets consisting of 3-tuples of the form (*X*, *Y*, *r*), where *X* and *Y* are object classes or instances and *r* is a set of primitive projection relations which semantically corresponds to their disjunction. Each pair (*X*,*Y*) must satisfy the corresponding relation set *r*. For example, the query expressed in the previous paragraph can be formally expressed as: Q_I = {(residential area, commercial center, {*southwest*}), (residential area, flood, {*covered_by*})}

During the execution of the query, stored images are sequentially examined and different instantiations of pairs of objects are assessed for matching each of the above tuples. Consider a generic query $Q=\{(X_i, Y_i, r_i) \mid i=1..n\}$ and a particular image instantiation $I = \{(X_{I\text{-}i}, Y_{I\text{-}i}, R_i) \mid i=1..n\}$. The similarity measure that we adopt is:

$$Similarity(I,Q)=\frac{\sum_{i=1}^{n}(Max\_Degree\_of\_Neighbourhood - d(R_i, r_i))}{n}$$

The maximum similarity is equal to *MAX_Degree_of _Neighbourhood* when images contain exactly the same query objects related by the same binary projection relations. The minimum similarity is 0 when none of the query objects can be instantiated, or in the extreme case where the distance between all instantiated relations and the corresponding query relations is *MAX_Degree_of_Neighbourhood*. We will illustrate image similarity using Figure 4, assuming one of the images to be the query image (*n*=10). The resultant similarity measure is 15.5 (the only differences between the images are caused by the movement of the flood).

Image I_1	Image I_2	*degree of neighbourhood*
$R_{12\text{-}1}(A_1,A_2)$	$R_{12\text{-}1}(A_1,A_2)$	*d*=0
$R_{13\text{-}1}(A_1, C_1)$	$R_{13\text{-}1}(A_1, C_1)$	*d*=0
$R_{11\text{-}1}(A_1, C_2)$	$R_{11\text{-}1}(A_1, C_2)$	*d*=0
$R_{3\text{-}8}(A_1,Flood)$	$R_{8\text{-}8}(A_1,Flood)$	*d*=2
$R_{13\text{-}3}(A_2, C_1)$	$R_{13\text{-}3}(A_2, C_1)$	*d*=0
$R_{3\text{-}1}(A_2, C_2)$	$R_{3\text{-}1}(A_2, C_2)$	*d*=0
$R_{1\text{-}11}(A_2,Flood)$	$R_{3\text{-}11}(A_2,Flood)$	*d*=2
$R_{1\text{-}3}(C_1, C_2)$	$R_{1\text{-}3}(C_1, C_2)$	*d*=0
$R_{1\text{-}13}(C_1,Flood)$	$R_{1\text{-}13}(C_1,Flood)$	*d*=0
$R_{2\text{-}13}(C_2,Flood)$	$R_{3\text{-}13}(C_2,Flood)$	*d*=1

$$Similarity(I_1, I_2)=\frac{16*10 - (2+2+1)}{10}=15.5$$

Figure 6 An example similarity assessment

Nabil et al. (1996) also propose a projection-based technique that uses conceptual neighbourhoods to measure image similarity. However they only deal with cases where images are re-arrangements of the same set of objects. The queries simply retrieve all images where some configuration of specific objects is satisfied (e.g., "find all images where object A is *above* object B…"). The general problem of configuration similarity where images contain arbitrary objects and the queries refer to object variables rather than instances is exponential to the number of objects in stored images because of possible multiple instantiations.

Applying the query {(residential area, commercial center, {*southwest*}), (residential area, flood, {*covered_by*})} to the images of Figure 4 we get two possible instantiations for the residential area object, and two instantiations for the commercial center object, resulting in a total of four instantiations for each image (Figure 7). Every instantiation produces a different sub-image with its own similarity. The user may impose a degree of acceptance, so the result consists of all subimages that have a difference from the target score less than a given degree. In the example of Figure 7, if the given degree is 1, the first sub-image of I_1, and the two first sub-images of I_2 will be returned to the user.

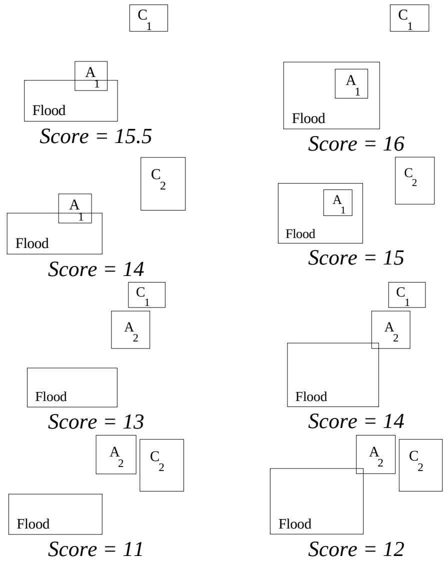

Figure 7 Example image similarity retrieval

The exponential structure of image similarity retrieval is problematic for applications involving large images. However in many situations a large number of instantiations produce differences larger than the target early in the search process. In this case we do not need to proceed with the rest of the objects, because even if their instantiations result in perfect relation matches, the target difference cannot be satisfied.

5. CONCLUSION

In this paper we discuss a form of spatial similarity based on relations. First, we introduce a generalised framework for the definition of projection-based relations in N-dimensional spaces. Topological, directional and qualitative distance relations can be expressed in various granularities and reasoning on conceptual neighbourhoods is significantly facilitated. On these grounds we develop a framework for handling object and image similarity queries which can manage uncertainty in the definition of spatial relations and fuzzy spatial objects' boundaries. Our results are important for most domains involving spatial data, such as GIS and CAD.

REFERENCES

Allen, J.F. "Maintaining Knowledge about Temporal Intervals". *CACM* 26 (11), pp. 832-843, 1983.

Bruns, T.H., Egenhofer, M.J. "Similarity of Spatial Scenes". *7th Symposium on Spatial Data Handling* (SDH), 1996.

Egenhofer, M.J., Al-Taha, K. "Reasoning about Gradual Changes of Topological Relations", *International Conference GIS- From Space to Territory*. Springer Verlag LNCS, 1992.

Egenhofer, M.J., Franzosa R., "Point Set Topological Relations", *IJGIS*, Vol. 5, pp. 161-174, 1991.

Freksa, C., "Temporal Reasoning based on Semi Intervals", *Artificial Intelligence*, Vol 54, pp. 199-227.

Nabil, M., Ngu, A., Shepherd, J. "Picture similarity retrieval using 2d projection interval representation". IEEE TKDE, 8(4), 1996.

Papadias, D., Egenhofer, M.J., "Algorithms for Hierarchical Spatial Reasoning", *Geoinformatica*, Vol. 1(3), to appear.

Papadias, D., Theodoridis, Y., Sellis, T., Egenhofer, M.J., "Topological Relations in the World of Minimum Bounding Rectangles: a Study with R-trees", *ACM SIGMOD*, 1995.

Salton, G., Allan, J., Buckley, C., "Automatic Structuring and Retrieval of Large Text Files". *CACM*, 37(2): 97-108, 1994.