

Similarity of Cardinal Directions*

Roop K. Goyal¹ and Max J. Egenhofer²

¹ ESRI, 380 New York Street, Redlands, CA 92373-8100, USA
rgoyal@esri.com

² National Center for Geographic Information and Analysis
Department of Spatial Information Science and Engineering
Department of Computer Science
Boardman Hall, University of Maine, Orono, ME 04469-5711, USA
max@spatial.maine.edu

Abstract. Like people who casually assess similarity between spatial scenes in their routine activities, users of pictorial databases are often interested in retrieving scenes that are similar to a given scene, and ranking them according to degrees of their match. For example, a town architect would like to query a database for the towns that have a landscape similar to the landscape of the site of a planned town. In this paper, we develop a computational model to determine the directional similarity between extended spatial objects, which forms a foundation for meaningful spatial similarity operators. The model is based on the direction-relation matrix. We derive how the similarity assessment of two direction-relation matrices corresponds to determining the least cost for transforming one direction-relation matrix into another. Using the transportation algorithm, the cost can be determined efficiently for pairs of arbitrary direction-relation matrices. The similarity values are evaluated empirically with several types of movements that create increasingly less similar direction relations. The tests confirm the cognitive plausibility of the similarity model.

1 Introduction

Similarity is an intuitive and subjective judgment, which displays no strict mathematical models (Tversky 1977). In their routine activities, people casually assess similarity between spatial scenes. Similarity also matters when users of spatial databases want to retrieve spatial scenes that resemble a sketched configuration and want to rank the results according to degrees of their match. Computational methods that match with users' intuitive expectations are the focus of this paper.

* This work was partially supported by the National Imagery and Mapping Agency under grant number NMA202-97-1-1023. Max Egenhofer's research is further supported by the National Science Foundation under grant numbers IRI-9613646, IIS-9970123, and EPS-9983432; by the National Imagery and Mapping Agency under grant NMA201-00-1-2009; by the National Institute of Environmental Health Sciences, NIH, under grant number 1 R 01 ES09816-01, and by a contract with Lockheed Martin M&DS.

A number of models and systems for spatial similarity retrieval have been already developed. For example, Query by Image Content (Flickner *et al.* 1995) allows a user to retrieve images from a database based on the contents of images. Graphic representations of images store geometric and visual attributes of objects and spatial relations between them. The geometric attribute of an object refers to its spatial extent, and visual attributes refer to color, shape, and texture (Gonzalez and Woods 1992). Geometric and visual attributes help in determining the presence of an object in a scene and spatial relations between objects distinguish relative placements of the objects in the embedding space. Combining object similarity and spatial relation similarity, one can make a query such as “Find scenes where object *A* and *B* are present, *B* is north of *A*, and *A* disjoint *B*.” Nabil *et al.* (1995) and Bruns and Egenhofer (1996) use spatial relations between objects for the assessment of scene similarity. Spatial relations are also used for similarity assessment in image databases (Chu *et al.* 1994; Del Bimbo *et al.* 1995; Del Bimbo and Pala 1997; Chu *et al.* 1998), multimedia databases (Al-Khatib *et al.* 1999; Yoshitaka and Ichikawa 1999), and video databases (Jiang and Elmagarmid 1998; Pissinou *et al.* 1998; Aslandogan and Yu 1999). In order to use spatial relations for similarity assessment, we need cognitively plausible methods to assess similarity between spatial relations.

Cardinal directions play an important role in the specification of spatial configurations. For example, the query scene (Figure 1a) and the three scenes in the database (Figures 1b-d) contain objects *A* and *B* whose topological relation is always disjoint; therefore, the three scenes are topologically equivalent to the query scene. However, when considering the *cardinal direction* as an additional search criterion, one can determine that Scene 1b is the most similar to the query scene. Also, when considering directions, one can determine that Scene 1c is more similar to the query scene than Scene 1d.

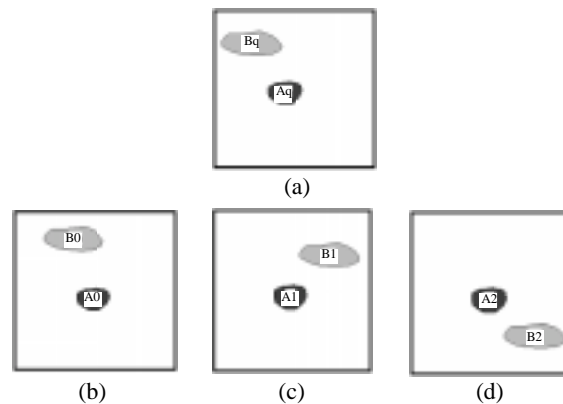


Fig. 1: (a) The query scene and (b)-(d) scenes 0, 1, and 2 in a database.

Earlier models for spatial-relation similarity resorted to minimum-bounding rectangles (Egenhofer 1997; Papadias and Delis 1997), representative points (Gudivada

and Raghavan 1995), or projections along the coordinate axes (Sistla *et al.* 1995) to model direction relations. These methods are crude approximations that often lead to incorrect directions when concave region objects are involved. The direction-relation (Goyal and Egenhofer, in press) overcomes these deficiencies as it avoids generalizations to points or approximations by rectangles. The method also extends to other geometric types such as lines and points (Goyal and Egenhofer 2000). This paper develops a computational method for assessing similarity of cardinal directions that are modeled with the detailed direction-relation matrix (Goyal and Egenhofer, in press). The better-quality direction-relation model displays some adverse properties that make it infeasible to apply the usual approaches for determining relation similarity; therefore, we introduce a new approach to determining spatial similarity that is based on the transportation algorithm. The method is cognitively plausible as well as computationally efficient.

The remainder of this paper is structured as follows. Following a short summary of the direction-relation matrix (Section 2), we introduce distance measures for pairs of simple (Section 3) and complex (Section 4) direction-relation matrices. Section 5 formulates the similarity assessment of two direction-relation matrices as a minimum-cost transformation. In Section 6 we demonstrate that the results of this similarity assessment method are cognitively meaningful by comparing the trends of similarity values obtained for moving objects with an expected baseline. Section 7 contains conclusions and discusses future work.

2 Direction-Relation Matrix

A cardinal direction is a binary relation involving a *reference object A* and a *target object B*, and a symbol that is a non-empty subset of {N, S, E, W, NE, SE, SW, NW, 0}. The symbols are spatially organized according to a grid formed around the target object (Figure 2), creating nine direction tiles (Figure 1): north (N_A), northeast (NE_A), east (E_A), southeast (SE_A), south (S_A), southwest (SW_A), west (W_A), northwest (NW_A), and same (0_A).

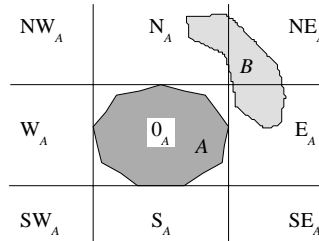


Fig. 2: The nine direction tiles formed around the reference object A.

The detailed direction-relation matrix (Goyal and Egenhofer, in press) is a 3×3 matrix that captures the neighborhood of the partition around the reference object and registers for each tile how much of the target object falls into it (Equation 1). The elements in the direction-relation matrix have the same topological organization as the partitions around the reference object.

$$dir(A,B) = \begin{bmatrix} \frac{\text{area}(\text{NW}_A \cap B)}{\text{area}(B)} & \frac{\text{area}(\text{N}_A \cap B)}{\text{area}(B)} & \frac{\text{area}(\text{NE}_A \cap B)}{\text{area}(B)} \\ \frac{\text{area}(\text{W}_A \cap B)}{\text{area}(B)} & \frac{\text{area}(0_A \cap B)}{\text{area}(B)} & \frac{\text{area}(\text{E}_A \cap B)}{\text{area}(B)} \\ \frac{\text{area}(\text{SW}_A \cap B)}{\text{area}(B)} & \frac{\text{area}(\text{S}_A \cap B)}{\text{area}(B)} & \frac{\text{area}(\text{SE}_A \cap B)}{\text{area}(B)} \end{bmatrix} \quad (1)$$

For example, the detailed direction-relation matrix for the configuration shown in Figure 1 has six elements of zero, and three non-zero elements which sum up to 1.0 (Equation 2)

$$dir(A,B) = \begin{bmatrix} 0 & 0.05 & 0.45 \\ 0 & 0 & 0.50 \\ 0 & 0 & 0 \end{bmatrix} \quad (2)$$

A direction-relation matrix with exactly one non-zero element is called a *single-element direction-relation matrix*, and the corresponding relation is a *single-element direction*. There are nine single-element directions corresponding to nine cardinal directions. A direction-relation matrix with more than one non-zero element is called a *multi-element direction-relation matrix*, and its corresponding direction is a *multi-element direction*. Subsequently, we develop a method to compute distances between single-element directions (Section 3), followed by an extension for multi-element directions (Section 4).

3 The Distance between Two Single-Element Direction-Relation Matrices

As a quantitative measure for direction similarity, we introduce a distance measure between two cardinal directions, such that (1) a zero value implies that both directions are identical and (2) $distance(D_0, D_1) > distance(D_0, D_2)$ means D_0 is more similar to D_2 than D_0 to D_1 , where D_0 , D_1 , and D_2 denote cardinal directions from B to A in Scenes 0, 1, and 2, respectively. A conceptual neighborhood graph for a set of mutually exclusive spatial relations serves as the basis for computing distances between the relations in this set. Conceptual neighborhood graphs have been used for deriving distance measures between 1-dimensional interval relations (Freksa 1992), topological relations in R^2 (Egenhofer and Al-Taha 1992), and relations between minimum bounding rectangles (Papadias and Delis 1997). A continuously changing relation follows a path along the conceptual neighborhood graph. For example, if a

target object *B* moves eastward from the northwest tile (Figure 3a) it cannot move directly to the northeast tile (Figure 3c), because it must go through the direction tile(s) that connect the northwest and northeast tiles. The shortest path would lead through the north tile (Figure 3b), although other connected paths are possible as well (e.g., through the west, same, and east tiles).

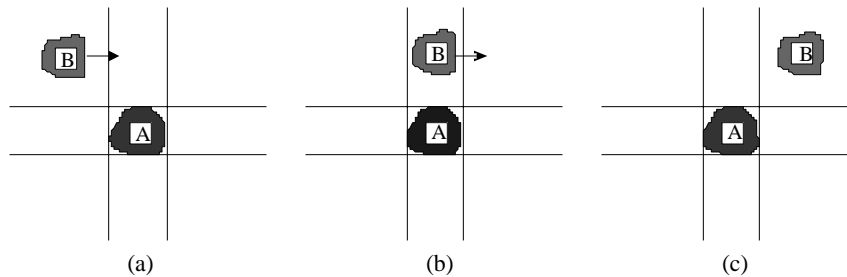


Fig. 3: The shortest path to move the target object *B* from the northwest tile to the northeast tile is through the north tile.

In order to compute the distance between cardinal directions, we construct a conceptual neighborhood graph for the nine cardinal directions using the 4-neighborhood of the nine tiles. This graph has a vertex for each cardinal direction and an edge for each pair of cardinal directions that are horizontally or vertically adjacent (Figure 4). The distance between two cardinal directions is the length of the shortest path between two directions along the conceptual neighborhood graph (Figure 5). The distance between two identical directions is zero, which is the shortest of all distances. The distance between the cardinal directions northwest and southeast is four, which is the maximum. The only other pair with the maximum distance is northeast and southwest. The distance function abides by the axioms of a distance—positivity, symmetry, and triangle inequality.

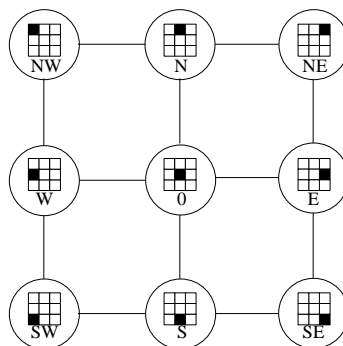


Fig. 4: The conceptual neighborhood graph for nine cardinal directions based on the 4-neighborhood between tiles.



















									
	N	NE	E	SE	S	SW	W	NW	0
 N	0	1	2	3	2	3	2	1	1
 NE	1	0	1	2	3	4	3	2	2
 E	2	1	0	1	2	3	2	3	1
 SE	3	2	1	0	1	2	3	4	2
 S	2	3	2	1	0	1	2	3	1
 SW	3	4	3	2	1	0	1	2	2
 W	2	3	2	3	2	1	0	1	1
 NW	1	2	3	4	3	2	1	0	2
 0	1	2	1	2	1	2	1	2	0

Fig. 5: Four-neighbor distances between cardinal directions for regions.

For example, the distance between the directions of B with respect to A along the conceptual neighborhood graph in Figures 3a and b is 1, while the distance between the directions in Figures 3a and c is 2. Based on these distances, we infer that the direction of B with respect to A in Figure 3a is more similar to the direction in Figure 3b than the direction in Figure 3a to the direction in Figure 3c.

The distance measure between single-element direction-relation matrices serves as the basis for the distance between multi-element direction-relation matrices.

4 Distance between Two Multi-Element Direction-Relation Matrices

To account for distances between multi-element direction-relation matrices we extend the method used for computing the distance between cardinal directions from single-element direction-relation matrices. If the target object B moves eastward from the northwest tile (Figure 6a) to the northeast tile (Figure 6e), its trajectory moves successively over the northwest and north tiles (Figure 6b), the north tile alone (Figure 6c), and the north and northeast tiles (Figure 6d). The directions in Figures 6b and d require multi-element direction-relation matrices for their representation.

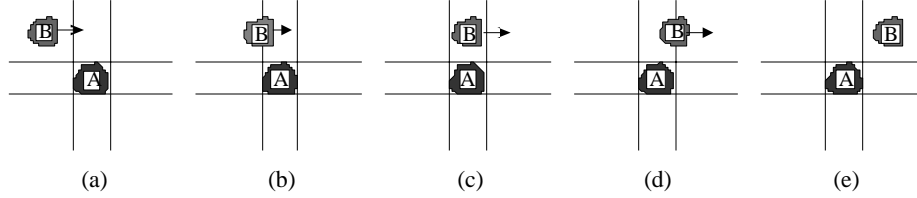


Fig. 6: The target object moves across single as well as multi-element cardinal directions from (a) northwest through (b) northwest and north, (c) north, (d) north and northeast, to (e) northeast.

Definition 1: The distance between two arbitrary direction-relation matrices, D^0 and D^1 , is the minimum cost for transforming matrix D^0 into D^1 by moving the non-zero elements of D^0 from their locations to the locations of the non-zero elements of D^1 along the conceptual neighborhood graph.

The cost of this transformation is the weighted sum of the distances along the neighborhood graph between the *source* and *destination* direction tiles, where a source refers to a cardinal direction from where a non-zero element is moved and a destination refers to a cardinal direction where the element is moved to. The weighting of a distance between a source and a destination is done by the element values moved between them. For example, transforming matrix D^0 (Equation 3a) into D^1 (Equation 3b) requires the movement of the value 0.4 from northwest to northeast, and the value 0.6 from north to northeast. The cost of this transformation is $0.4 \times \text{distance}(\text{NW}, \text{NE}) + 0.6 \times \text{distance}(\text{N}, \text{NE})$, which is $0.4 \times 2 + 0.6 \times 1 = 1.4$.

$$D^0 = \begin{bmatrix} 0.4 & 0.6 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3a)$$

$$D^1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3b)$$

The remainder of this section introduces consistency constraints and properties of intermediate matrices, which are needed to develop the method for distance computation between arbitrary direction-relation matrices.

Definition 2: The *sum* of a matrix P is defined as the sum of the values of its elements (Equation 4).

$$\text{sum}(P) := \sum_{\forall i} \sum_{\forall j} P_{i,j} \quad (4)$$

Definition 3: The *commonality* C^{01} between two direction-relation matrices, D^0 and D^1 , is a 3x3 matrix defined as the minimum of each pair of corresponding element values (Equation 5).

$$\forall i, j : C_{ij}^{01} \doteq \min(D_{ij}^0, D_{ij}^1) \quad (5)$$

The values of the elements in C^{01} lie in the interval $[0, 1]$. The value of $\text{sum}(C^{01})$ also lies in the interval $[0, 1]$. It is 0 if all the corresponding pairs of elements have at least one 0. It would be 1 if the distribution of the target objects in both D^0 and D^1 were identical. Since the minimum of a set of numbers is unique and does not depend on their order, the calculation of the commonality is commutative (Equation 6).

$$C^{01} = C^{10} \quad (6)$$

For example, the commonality matrices C^{01} and C^{10} for directions of B with respect to A in Scene 0 and Scene 1 (Figure 7) have the same values (Equation 7).

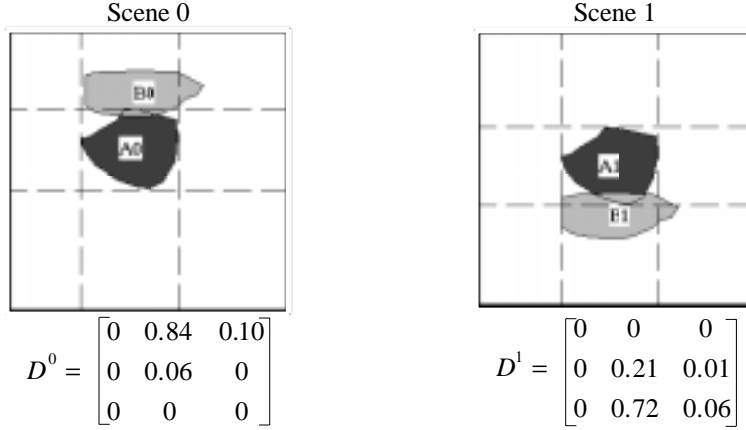


Fig. 7: A direction comparison of Scenes 0 and 1 with identical objects, but different directions D^0 and D^1 , respectively.

$$C^{01} = C^{10} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.06 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (7)$$

Definition 4: The *asymmetric difference* R^{01} between two direction-relation matrices, D^0 and D^1 , is defined as the difference of the direction-relation matrix D^0 and the commonality (Equation 8a), and has a corresponding value for R^{10} (Equation 8b).

$$R^{01} \doteq D^0 - C^{01} \quad (8a)$$

$$R^{10} \doteq D^1 - C^{10} \quad (8b)$$

We use the term *non-zero part of a matrix* for a non-zero element or a fraction of a non-zero element. The asymmetric difference R^{01} has the distinct *non-zero parts* of D^0 that are not in D^1 . Conversely, R^{10} (Equation 8b) has the distinct non-zero parts of D^1 that are not in D^0 . For example, the asymmetric difference matrix R^{01} (Equation 9a)

for the scenes in Figure 7 has no non-zero parts that are in D^1 , while R^{10} (Equation 9b) has no non-zero parts that are in D^0 .

$$R^{01} = \begin{bmatrix} 0 & 0.84 & 0.10 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (9a)$$

$$R^{10} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.15 & 0.01 \\ 0 & 0.72 & 0.06 \end{bmatrix} \quad (9b)$$

The values of elements in R^{01} and R^{10} lie in the closed interval $[0, 1]$. The values of $sum(R^{01})$ and $sum(R^{10})$ also lie in the interval $[0, 1]$. The value 0 for $sum(R^{10})$ means there is no difference between matrices D_0 and D_1 , whereas the value 1 means there is no commonality between matrices D_0 and D_1 .

Definition 5: The *direction-difference* (Δ^{01}) between two direction-relation matrices, D^0 and D^1 , is defined as the difference of the two relations' asymmetric differences (Equation 10).

$$\Delta^{01} := R^{01} - R^{10} \quad (10)$$

The values of the elements in Δ^{01} lie in the interval $[-1, 1]$. In order to express the direction-difference in terms of direction-relation matrices, we substitute the value of R^{01} and R^{10} (Equations 8a and b) in Equation 10, and obtain Equation 11, as commonality matrices C^{01} and C^{10} cancel each other. For example, Equation 12 gives the direction-difference matrices Δ^{01} for the scenes in Figure 7.

$$\Delta^{01} = D^0 - D^1 \quad (11)$$

$$\Delta^{01} = \begin{bmatrix} 0 & 0.84 & 0.10 \\ 0 & -0.15 & -0.01 \\ 0 & -0.72 & -0.06 \end{bmatrix} \quad (12)$$

Theorem 1: The sum of the elements in R^{01} equals the sum of the elements in R^{10} .

Proof: The sum of the elements of a detailed direction-relation matrix is 1 (Equation 13).

$$sum(D^0) = sum(D^1) = 1 \quad (13)$$

The addition operation (+) follows the associative law; therefore, we can express the asymmetric difference matrices (Equations 8a-b) in the sum form (Equations 14a-b).

$$sum(R^{01}) = sum(D^0) - sum(C^{01}) \quad (14a)$$

$$sum(R^{10}) = sum(D^1) - sum(C^{10}) \quad (14b)$$

Let us assume the value of $sum(C^{01})$ is x , which equals $sum(C^{10})$; substituting x for these sums in Equation 14a-b and combining them with Equation 13, we obtain expressions for the sum of asymmetric difference matrices in terms of x (Equations 15a-b).

$$sum(R^{01}) = 1 - x \quad (15a)$$

$$sum(R^{10}) = 1 - x \quad (15b)$$

The right hand sides of both asymmetric difference matrices' sums are identical (Equations 15a-b), which proves the theorem (Equation 16).

$$sum(R^{01}) = sum(R^{10}) \quad (16)$$

∴

Corollary 2: The sum of the elements in a direction-difference matrix is 0.

Proof: We can express the direction-difference matrix (Equation 10) in the sum form (Equation 17). Since the values of the sums of the asymmetric difference matrices' elements are identical (Equation 16), their differences cancel each other (Equation 18).

$$sum(\Delta^{01}) = sum(R^{01}) - sum(R^{10}) \quad (17)$$

$$sum(\Delta^{01}) = 0 \quad (18)$$

∴

For example, the sum of elements of Δ^{01} (Equation 12) is zero. Non-zero elements of the commonality matrix C^{01} capture the common non-zero parts of the matrices D^0 and D^1 ; therefore, the non-zero parts that correspond to non-zero parts of C^{01} must not be moved while transforming D^0 into D^1 (Definition 1). Moving them would increase the cost of this transformation, such that the computed cost for this transformation would not be the minimum value. Only those non-zero parts of D^0 should be moved that are zero in D^1 , which means only non-zero elements of R^{01} must be moved to obtain R^{10} .

Definition 6: The distance between two matrices D^0 and D^1 is the minimum cost incurred in transforming R^{01} into R^{10} .

Theorem 3: The maximum 4-neighbor distance ($distance_{max}^4$) between two direction-relation matrices is 4.

Proof: The maximum cost is incurred when the maximum possible value of $sum(R^{01})$ is moved by the maximum possible distance. The maximum value of $sum(R^{01})$ is 1 and the maximum 4-neighbor distance between two cardinal directions is 4 (Figure 5); therefore, the value of $distance_{max}^4$ is $4 \times 1 = 4$. ∴

The maximum distance between two direction-relation matrices can occur only between single-element direction-relation matrices that have non-zero values for the farthest direction tiles. For example, the value of 4-neighbor distance between two single-element direction-relation matrices with non-zero values in the southwest and northeast tiles is 4. In the direction-difference Δ^{01} , non-zero elements that correspond

to non-zero elements of R^{01} are of positive polarity, while non-zero elements that correspond to non-zero elements of R^{10} are of negative polarity (Equation 10). The sum of the elements of the matrix Δ^{01} is zero (Corollary 2). The matrix Δ^{01} has all the necessary information to compute the minimum cost of transforming D^0 into D^1 , which is the same as the minimum cost of transforming R^{01} into R^{10} .

5 Minimum Cost Solution

The problem of determining the minimum cost for transforming matrix R^{01} into matrix R^{10} can be formulated as a balanced *transportation problem* (Murty 1976; Strayer 1989), which is a special case of the linear programming problem (Dantzig 1963; Dantzig and Thapa 1997). A transportation problem records the supplies of all the *warehouses*, the demands of all the *markets*, and the *unit costs* for all the pairs of the warehouses and the markets. This setting applies to the direction difference matrix Δ^{01} , where each positive element in Δ^{01} corresponds to a warehouse, whereas each negative element corresponds to a market. The supply of the i th warehouse W_i is s_i , which equals the magnitude of the corresponding element in Δ^{01} . Similarly, the demand of the j th market M_j is d_j , which also equals the magnitude of the corresponding element in Δ^{01} . The cost c_{ij} for moving a unit supply from W_i to M_j is *distance* (W_i, M_j). The *sum* (R^{01}) equals the *sum* (R^{10}) (Theorem 1); therefore, the sum of the supplies of all warehouses is the same as the sum of the demands of all markets, which identifies a balanced transportation problem.

If x_{ij} is the number of units to be shipped from the warehouse W_i to the market M_j in the final solution, then the goal of the transportation problem is to determine the values of x such that the total cost z (Equation 19) is a minimum such that the warehouse and market constraints are satisfied (Equations 20a-b).

$$z = \sum_{i=1}^p \sum_{j=1}^n c_{ij} x_{ij} \quad (19)$$

$$\forall i \in W : s_i = \sum_{j=1}^n x_{ij} \quad (20a)$$

$$\forall j \in M : d_j = \sum_{i=1}^p x_{ij} \quad (20b)$$

While each constraint can be formulated as a linear equation, the system of the warehouse and market constraints is underdetermined in most cases, such that the common algebraic methods for linear equations do not provide a unique solution. For the direction difference with nine elements, a total of twenty different scenarios may occur. In the worst case, there are $4*5=20$ unknown elements with $2+4=6$ warehouse and market constraints (Equations 20a and b) and one additional constraint for the minimum cost (Equation 19). Other scenarios may be less underconstrained—the

direction difference for the example in Figure 7 has eight unknown variables with seven constraints total—or potentially overconstrained as in the case of a direction difference with 1 warehouse and n markets. To accommodate for these variations, typically the transportation problem is solved in two phases: (1) finding a basic feasible solution and (2) improving the basic feasible solution iteratively, until an optimal solution is obtained.

A basic feasible solution is a set of x values that satisfy the market and warehouse constraints, but it may not give the minimum value of z . There can be more than one set of x values that yield the same minimum value z . A common algorithm to determine a basic feasible solution is the *northwest¹ corner method* (Strayer 1989, p. 180), while the minimum can be obtained through the *transportation algorithm* (Strayer 1989, p. 153). Since both algorithms are classic, we do not repeat them here but refer the interested reader either to a textbook (Strayer 1989) or the details of their use for calculating direction similarity (Goyal 2000, pp. 100-108).

The example in Figure 7 formulates 2+4=6 warehouse and market constraints (Equations 21a-f) and the cost function (Equation 21g).

$$x_{11} + x_{12} + x_{13} + x_{14} = 0.84 \quad (21a)$$

$$x_{21} + x_{22} + x_{23} + x_{24} = 0.10 \quad (21b)$$

$$x_{11} + x_{21} = 0.06 \quad (21c)$$

$$x_{12} + x_{22} = 0.72 \quad (21d)$$

$$x_{13} + x_{23} = 0.01 \quad (21e)$$

$$x_{14} + x_{24} = 0.15 \quad (21f)$$

$$z = 3x_{11} + 2x_{12} + 2x_{13} + x_{14} + 2x_{21} + 3x_{22} + x_{23} + 2x_{24} \quad (21g)$$

The northwest corner method offers then a basic feasible solution at $z_{\text{feasible}}=1.89$ (Equation 22a), for which the transportation algorithm finds a minimum at $z_{\text{minimum}}=1.75$ (Equation 22b).

$$z = 3 * 0.06 + 2 * 0.72 + 2 * 0.01 + 1 * 0.05 + 2 * 0.10 \quad (22a)$$

$$z = 2 * 0.72 + 1 * 0.12 + 2 * 0.06 + 1 * 0.01 + 2 * 0.03 \quad (22b)$$

Although the simplex method has a non-polynomial upper bound, it requires an average of $3m$ pivot steps, where m is the number of constraints in the problem. For the 3×3 direction-relation matrix the largest value for m is 9, therefore, the simplex method can be performed in at most 27 pivot steps. In comparison, relying on a graph

¹ The term *northwest corner* refers here to the top-left cost value, that is, the first row and first column in the transportation tableau used to solve the transportation problem. It should not be confused with the *northwest tile* in the direction-relation matrix.

of all possible movements could result in a graph with as many as 17,250,360 paths (Goyal 2000). Therefore, the method based on the transportation algorithm is significantly more efficient.

Definition 7: The *dissimilarity* of two pairs of cardinal directions expressed in the form of two direction-relation matrices D^0 and D^1 , corresponds to the minimum cost of transforming D^0 into D^1 , normalized by the highest possible cost (Equation 23).

$$dissimilarity(D^0, D^1) = \frac{minCost(D^0, D^1)}{4} \quad (23)$$

Definition 8: The similarity s of two pairs of cardinal directions is the complement of the dissimilarity (Equation 24).

$$s(D^0, D^1) = 1 - dissimilarity(D^0, D^1) \quad (24)$$

Applying the similarity assessment method to the query scene in Figure 1a and Scenes 0-2 in the database (Figures 5.1b-d), the similarities values are $s(D^q, D^0) = 0.92$, $s(D^q, D^1) = 0.56$, and $s(D^q, D^2) = 0.04$. From these values, we can infer that the direction in the query scene is most similar to the direction in Scene 0, followed by Scene 1, and Scene 2, as the similarity is decreasing in this order.

6 Evaluation

The subsequent benchmark consists of a set of successive changes in the geometry of the involved objects, making deformations that create an increasingly less similar situation. If the similarity values capture appropriately the decrease in similarity, then these results provide an indication that the similarity values are semantically meaningful.

For a systematic evaluation of the method for similarity assessment, we start with a query scene containing a pair of reference and target objects, and generate scenes by gradually changing the geometry of the target object. Throughout these changes, we capture the direction similarity between the query scene and the test cases and examine the trend of the similarity values. A cognitively plausible similarity assessment method is expected to give the highest similarity value (100) to the scene without any change and continuously decreasing values of similarity with increasing changes.

6.1 Moving the Target Object over the Reference Object

In this test, the target object B is moved iteratively from the northeast of A to the southwest of A . The expected graph is a straight line from the top left (100) to the bottom right (0). The eight different states and their corresponding similarity values with respect to the query scene (Figure 8) lead to a strictly monotonic decrease in similarity (Table 1a). The minor deviations from the expected graph are due to varying increments at which the snapshots were taken.

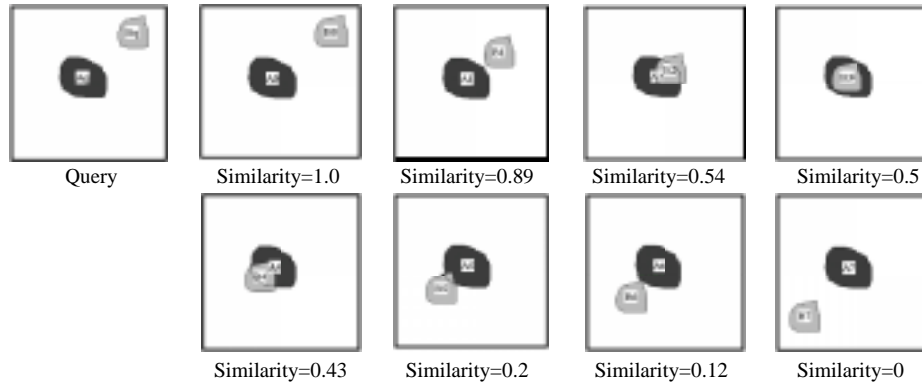


Fig. 8: Detailed diagonal movement of a target object, together with the corresponding similarity values.

Two similar assessments confirm this pattern and provide additional insights about the properties of the direction-similarity assessment. Figures 9 and 10 show two diagonal movements of objects of different sizes—B larger than A, and A larger than B, respectively—recorded at coarser levels of detail than in Figure 8. The resulting graphs (Table 1b and 1c) have the same strictly monotonic decline; therefore, the similarity assessment shows the same trend for movement, independent of the objects’ sizes as well as the increments at which they are recorded.

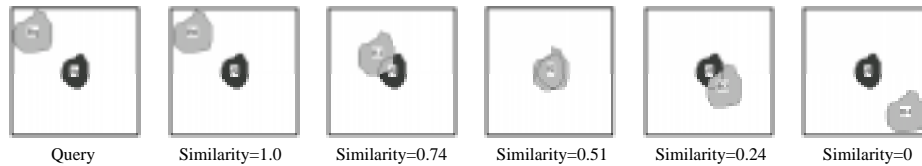


Fig. 9: Coarse diagonal movement of a larger target object, together with the corresponding similarity values for direction relations.

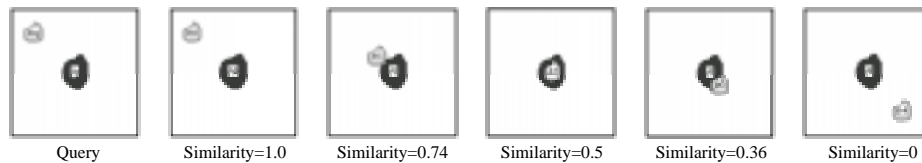


Fig. 10: Coarse diagonal movement of a smaller target object, together with the corresponding similarity values for direction relations.

6.2 Rotating the Target Object around the Reference Object

In the second set of tests, the target object B located in the northwest tile is rotated around the reference object A without intersecting with A. The expected pattern for this full-circle rotation is a symmetric v-shaped graph, where the highest dissimilarity is obtained when the target object has been rotated by 180°.

The 17 states with their corresponding similarity values (Figure 9) lead to a strictly monotonic decrease in similarity, followed by a strictly monotonic increase (Table 1d). This pattern also confirms that the same trend would be observed if the orientation of the rotation changed from clockwise to counterclockwise.



Fig. 11: Clockwise, full-circle rotation of the target object around the reference object, together with the corresponding similarity values for direction relations.

Two variations of this test confirm the general trend. The sequences in Figure 12 show a half-cycle rotation where A and B overlap partially. The expected graph is a monotonically falling line, without reaching 0 at the end, because the query and the last scene differ in less than the maximum distance. Table 1e confirms this expectation.

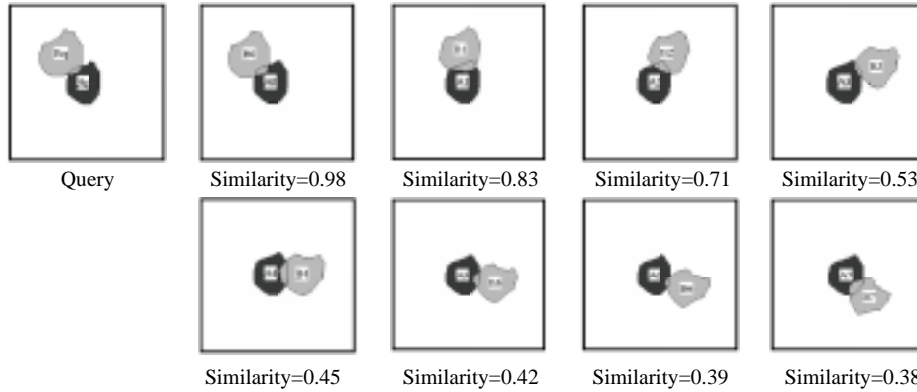


Fig. 12: Clock-wise, semi-circle rotation of a target object around a reference object, together with the corresponding similarity values for direction relations

Another type of rotation is examined by spinning an elongated object *B* around an axis such that some of *B* always remains to the northwest of the reference object *A*, while parts of *B* move across *A* (Figure 13). The expected similarity profile is the characteristic v-shape of a full-circle rotation (Table 1d) with a low point around 50%, however, because *B* does not make a full-circle transition to the opposite end of *A*'s northwest corner. The graph (Table 1f) confirms this behavior of the similarity measure.

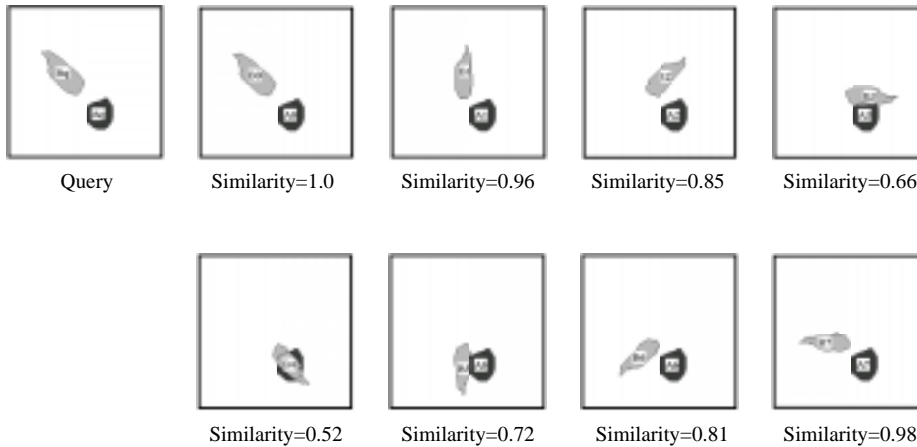


Fig. 13: Clockwise 360° rotation of a target object, together with the corresponding similarity values for direction relations

6.3 Scaling Up the Target Object

A third set of test cases analyzes the effects of scaling up the reference target B such the B's direction with respect to A is affected. First B is in the northwest tile and grows continuously (Figures 14), and then the same process is repeated with B starting due north of A (Figure 15). In both cases the anticipated changes in direction are monotonic, which are confirmed through the similarity graphs (Table 1g and h).

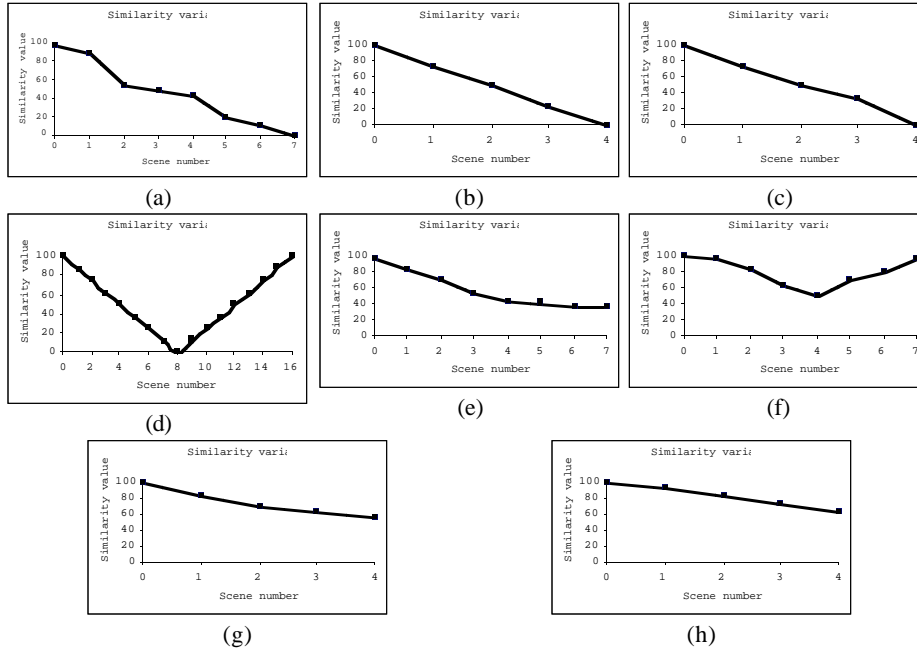


Fig. 14: Scaling up a target object located in the northwest tile of the reference object



Fig. 15: Scaling up a target object located in the north tile of the reference object.

Table 1. The direction profiles of the movements for diagonal movements (a-c), rotations (d-f), and scalings (g-h).



7 Conclusions and Future Work

The similarity assessment for direction-relation matrices corresponds to cognitively plausible cases, and provides a computationally efficient method via the transportation algorithm. Unlike other direction-relation representations, the direction-relation matrix provides an accurate representation of direction relations without resorting to such crude approximations as centroids or minimum-bounding rectangles. The computational model for assessing direction-relation similarity gave new insights for methods that may be useful in such settings as sketch-based retrieval of spatial configurations.

While we demonstrated through commonsensical evaluations the plausibility of the approach, it might be of interest to investigate other methods for determining the distances within the conceptual neighborhood graph among tiles. Goyal (2000) already demonstrated that the use of the 8-neighborhood of direction tiles in lieu of the 4-neighborhood leads to counterintuitive similarity values. Are there other neighborhood configurations that would give better results than the 4-neighborhood? Additional evaluations with human subject tests will be needed to confirm the concordance with people's judgments so that meaningful spatial operators for query languages can be obtained. The overall method for determining similarity of direction

relations is flexible enough to accommodate such variations in the distances between single-element direction relations.

A significant side product of this method of similarity assessment is the new insight that sequences of direction-similarity values for continuously moving objects may be used as a high-level description of the objects' movements. These sequences not only capture the closest relations, but also provide characteristic profiles for high-level spatial concepts. For example, a particular graph may link directly to a type of deformation. Therefore, similarity profiles of the sort presented in Table 1 may capture high-level types of spatial changes, which then translate into meaningful cognitive or linguistic spatial concepts. In particular, the combined assessment of such direction profiles with topological profiles (Egenhofer and Al-Taha, 1992) appears to be very productive.

8 References

- W. Al-Khatib, Y. Day, A. Ghafoor, and P. Berra (1999) Semantic Modeling and Knowledge Representation in Multimedia Databases. *IEEE Transactions on Knowledge and Data Engineering* 11(1): 64-80.
- Y. Aslandogan and C. Yu (1999) Techniques and Systems for Image and Video Retrieval. *IEEE Transactions on Knowledge and Data Engineering* 11(1): 56-63.
- T. Bruns and M. Egenhofer (1996) Similarity of Spatial Scenes. in: M.-J. Kraak and M. Molenaar (eds.), *Seventh International Symposium on Spatial Data Handling*, Delft, The Netherlands, pp. 173-184.
- W. Chu, C. Hsu, A. Cardenas, and R. Taira (1998) Knowledge-based Image Retrieval with Spatial and Temporal Constructs. *IEEE Transactions on Knowledge and Data Engineering* 10(6): 872-888.
- W. Chu, I. Leong, and R. Taira (1994) A Semantic Modeling Approach for Image Retrieval by Content. *VLDB Journal* 3(4): 445-477.
- G. Dantzig (1963) *Linear Programming And Extensions*. Princeton University Press, Princeton, NJ.
- G. Dantzig and M. Thapa (1997) *Linear Programming*. Springer-Verlag, New York.
- A. Del Bimbo and P. Pala (1997) Visual Image Retrieval by Elastic Matching of User Sketches. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(2): 121-132.
- A. Del Bimbo, E. Vicario, and D. Zingoni (1995) Symbolic Description and Visual Querying of Image Sequences Using Spatio-Temporal Logic. *IEEE Transactions on Knowledge and Data Engineering* 7(4): 609-622.
- M. Egenhofer (1997) Query Processing in Spatial-Query-by-Sketch. *Journal of Visual Languages and Computing* 8(4): 403-424.
- M. Egenhofer and K. Al-Taha (1992) Reasoning About Gradual Changes of Topological Relationships. in: A. Frank, I. Campari, and U. Formentini (eds.), *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Pisa, Italy, Lecture Notes in Computer Science, 639: 196-219.
- M. Flickner, H. Sawhney, W. Niblack, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker (1995) Query by Image and Video Content: The QBIC System. *IEEE Computer* 28(9): 23-32.

- C. Freksa (1992) Temporal Reasoning Based on Semi-Intervals. *Artificial Intelligence* 54: 199-227.
- R. Goyal (2000) *Similarity Assessment for Cardinal Directions Between Extended Spatial Objects*. Ph.D. Thesis, Department of Spatial Information Science and Engineering, University of Maine, Orono, ME, http://www.spatial.maine.edu/Publications/phd_thesis/Goyal2000.pdf
- R. Goyal and M. Egenhofer (in press) Cardinal Directions between Extended Spatial Objects. *IEEE Transactions in Knowledge and Data Engineering* (in press).
- R. Goyal and M. Egenhofer (2000) Consistent Queries over Cardinal Directions across Different Levels of Detail. in: A.M. Tjoa, R. Wagner, and A. Al-Zobaidie (eds.), *11th International Workshop on Database and Expert Systems Applications*, Greenwich, U.K. pp. 876-880.
- R. Gonzalez and R. Woods (1992) *Digital Image Processing*. Addison-Wesley Publishing Company, Reading, MA.
- V. Gudivada and V. Raghavan (1995) Design and Evaluation of Algorithms for Image Retrieval by Spatial Similarity. *ACM Transactions on Information Systems* 13(2): 115-144.
- H. Jiang and A. Elmagarmid (1998) Spatial and Temporal Content-based Access to Hypervideo Databases. *VLDB Journal* 7: 226-238.
- K. Murty (1976) *Linear and Combinatorial Programming*. John Wiley & Sons, Inc., New York.
- M. Nabil, J. Shepherd, and A. Ngu (1995) 2D Projection Interval Relationships: A Symbolic Representation of Spatial Relationships. in: M. Egenhofer and J. Herring (eds.), *Advances in Spatial Databases—4th International Symposium, SSD '95, Portland, ME*, Lecture Notes in Computer Science, 951: 292-309 Springer-Verlag, Berlin.
- D. Papadias and V. Delis (1997) Relation-Based Similarity. *Fifth ACM Workshop on Advances in Geographic Information Systems*, Las Vegas, NV, pp. 1-4.
- N. Pissinou, I. Radev, K. Makki, and W. Campbell (1998) A Topological-Directional Model for the Spatio-Temporal Composition of the Video Objects. in: A. Silberschatz, A. Zhang and S. Mehrotra (eds.), *Eighth International Workshop on Research Issues on Data Engineering, Continuous-Media Databases and Applications*, Orlando, FL, pp. 17-24.
- A. P. Sistla, C. Yu, C. Liu, and K. Liu (1995) Similarity based Retrieval of Pictures Using Indices on Spatial Relationships. in: U. Dayal, P. Gray, and S. Nishio (eds.), *21st International Conference on Very Large Data Bases*, Zurich, Switzerland, pp. 619-629.
- J. Strayer (1989) *Linear Programming and Its Applications*. Springer-Verlag, New York.
- A. Tversky (1977) Features of Similarity. *Psychological Review* 84(4): 327-352.
- A. Yoshitaka and T. Ichikawa (1999) A Survey on Content-based Retrieval for Multimedia Databases. *IEEE Transactions on Knowledge and Data Engineering* 11(1): 81-93.