Study on Spatial Geometric Similarity Based on Conformal Geometric Algebra

Xiaomin Jiang 1,2, Yangfei Huang 1,2,* and Feng Zhang 3,4

1 School of Civil Engineering & Architecture, Zhejiang University of Science & Technology, 318 Liuhe Road, Hangzhou 310023, China
2 Center of Urban and Rural Development, Zhejiang University of Science & Technology, 318 Liuhe Road, Hangzhou 310023, China
3 Zhejiang Provincial Key Laboratory of Geographic Information Science, Zhejiang University, 148 Tianmushan Road, Hangzhou 310028, China
4 School of Earth Sciences, Zhejiang University, 38 Zheda Road, Hangzhou 310027, China
* Correspondence: huangyangfei@zust.edu.cn

Abstract: The study of spatial geometric similarity plays a significant role in spatial data retrieval. Many researchers have examined spatial geometric similarity, which is useful for spatial analysis and data retrieval. However, the majority of them focused on objects of the same type. Methods to support the spatial geometric similarity computation for different types of objects are rare, a systematic theory index has not been developed yet, and there has not been a comprehensive computational model of spatial geometric similarity. In this study, we conducted an analysis of the spatial geometric similarity computation based on conformal geometric algebra (CGA), which has certain advantages in the quantitative computation of the measurement information of spatial objects and the qualitative judgment of the topological relations of spatial objects. First, we developed a unified expression model for spatial geometric scenes, integrating shapes of objects and spatial relations between them. Then, we established a model for the spatial geometric similarity computation under various geographical circumstances to provide a novel approach for spatial geometric similarity research. Finally, the computation model was verified through a case study. The study of spatial geometric similarity sheds light on spatial data retrieval, which has scientific significance and practical value.

Keywords: conformal geometric algebra; shape similarity; spatial relations similarity; spatial geometric similarity

1. Introduction

With the diversification of Earth observation technology and the popularization of mobile devices, the amount and types of data related to geospatial positions are increasing rapidly. Determining how to locate spatial geometric scenes in which users are interested from massive spatial geometric scene data is a major issue for spatial data retrieval in urban scenarios. Computing the geometric similarity between a database scene and a query scene in a spatial database is a key point in solving the problem of spatial scene retrieval. The traditional spatial geometric similarity indices are the shape similarity of spatial objects, topological relation similarity, direction relation similarity, and distance relation similarity between spatial objects, and the computation methods are not uniform. The existing spatial data retrieval methods that were developed without comprehensive consideration of the above similarity indices cannot satisfy the demand for spatial geometric scene retrieval.

In 2001, Chinese mathematician Li Hongbo proposed the theory of conformal geometric algebra (CGA), and it was introduced to the geoscience field. This theory can realize the unified expression of different types and dimensions of geographic objects and has certain advantages in the quantitative computation of the measurement information of spatial objects and the qualitative judgment of the topological relations of spatial objects [1].
In this study, we focused on the problem of spatial geometric similarity. Based on CGA theory, a discussion is presented from the perspective of the shape expressions of spatial objects and similarity computation, the expression of the spatial relations and similarity computation, and the computation of the geometric similarity of a spatial scene. Finally, a set of relatively complete spatial geometric similarity retrieval models is proposed, which can satisfy the demand of qualitative spatial data retrieval to a certain degree.

The main contributions of the present study are as follows.

1. Based on CGA, we realize a unified representation of spatial geometric scenes with objects of different types. The spatial geometric scene we consider is composed of different types of geometric objects and is not restricted to polygons.

2. We perform geometric similarity computations between two spatial geometric scenes, taking the shape, topology, direction, and distance into consideration to attempt to implement spatial geometric retrieval in a comprehensive manner.

2. Literature Review

A spatial scene comprises a collection of spatial objects and their particular spatial arrangement. People construe such spatial geometric scenes in terms of objects and relations [2,3]. Many scholars have explored this topic [4–7]. Spatial scene similarity is the similarity of two regions in terms of particular object sizes and contents [5–8]. The spatial geometric similarity in our study involves spatial geometric features, including the shape of spatial objects and the spatial relationship between objects. The existing spatial similarity research is mainly divided into two parts. One is based on the geometric characteristics of spatial objects, and the similarity of spatial objects is studied. Kuijpers et al. [9] and Gottfried [10] used a double cross model and bipartite arrangements, respectively, to obtain qualitative shape expressions and proposed concrete qualitative shape similarity computation algorithms from the perspective of shape description operations. Shapiro et al. [11] proposed a theoretical relational modeling technique to separate objects into thin pieces (e.g., sticks, plates, and blobs). Richards et al. [12] used a codon as the basic unit to describe closed two-dimensional (2D) shapes.

The other approach uses the relations between spatial objects to study the similarity of the spatial relations [13–15]. Spatial relation similarity mainly includes spatial topological relation similarity, direction relation similarity, and distance relation similarity. Many scholars have performed some research on the combination of topology and direction, the combination of direction and distance, and the combination of the three and developed similarity evaluation methods, such as feature matching, derivation from equivalence, case reasoning, and fuzzy functions [16–21]. The semantic similarity plays an important role in spatial scene retrieval. Many researchers have made some contributions in this area [22–25].

Geometric algebra is an algebraic language that was proposed by Clifford [26] based on the Hamilton Quaternion and Grassmann expansion algebra to describe and perform computations for geometric problems. The unification of vector, scalar, dimension, and geometric operations makes it possible to apply geometric algebra for the expression of geographic objects and compute the spatial relations. The research on the application of geometric algebra has matured around the world [27–30], and there are various research results in the theoretical physics [31,32], mathematics [33,34], computer science [35], and geoscience [36–48] fields.

Li et al. [1] first presented the concept of CGA in the book Geometric Computing with Clifford Algebras, which has been highly praised by the international academic community. Zong [47] obtained a formal expression of the spatial topological relations based on algebraic geometry and the quantitative computation of topological relations by using a meet operator, which was able to distinguish more detailed spatial topological relations, and applied it to the intersection of the 3D surface model of an Antarctic ice sheet. Chen [48] obtained a unified expression of different types of multi-dimensional geographic objects using CGA properties, based on which formal expressions of the spatial topological relations and temporal topological relations were obtained. Furthermore, spatiotemporal
topological relations were obtained. These relations were applied to the computation of spatiotemporal topological relations. This approach has value for various applications, such as checking the spatiotemporal topological rules of current land use and the historical backtracking of spatiotemporal objects.

3. Methodology

In this paper, we introduce a multidimensional unified expression and geometric computation based on CGA. The unified expression of multidimensional geometric objects is realized based on the multivector. By using the geometric product, the similarity relationship of spatial objects, spatial topology, direction, and distance can be conveniently computed, which provides a new solution for the spatial geometric similarity retrieval.

According to Figure 1, first, we resolve the central objects that satisfy a specified shape similarity threshold. Second, the Voronoi graph is built to obtain the spatial neighborhood objects. From this, we can obtain a certain number of scenes. Third, according to the spatial constraints for the query scene, the variable ordering is set, and the forward checking algorithm is used to match the query scene with the sub-scenes from the database scene. If no such sub-scene meets the condition, we can relax the condition and compute the similarity between the query scene and the retrieved scene until we achieve our goal. The main theories covered in this paper are described in this section as follows.

![Figure 1. Spatial geometric similarity retrieval process.](chart)

3.1. Conformal Geometric Algebra

CGA is a new geometric representation and computing system that provides a unified and simple homogeneous algebraic framework for classical geometry and has notable advantages in geometric data processing and geometric computation [49]. The inner product, outer product, and geometric product are core operations in the geometric object representation and algorithm construction. CGA is the most common geometric algebra (GA) space. Based on the traditional Euclidean space, two extra base vectors, $e_0$ and $e_i$, are used to represent basic geometric objects. A hierarchical geometric structure corresponds
to the hierarchical Grassmann structure for the outer product in CGA. The inner product in CGA can describe basic metric information, such as the distance and angle.

3.2. Shape Similarity Computation

Falomir et al. [50] conducted similarity measurements of spatial objects based on qualitative shape descriptions. The complete description of the shape of a 2D object is given from a set of qualitative features as follows:

\[ EC_1, A_1 | TC_1, L_1, C_1 \] , \[ EC_2, A_2 | TC_2, L_2, C_2 \] , \ldots , \[ EC_n, A_n | TC_n, L_n, C_n \]

where \( n \) is the total number of relevant points of the object, \( EC_i \) describes the edge connection that occurs at the point \( P_i \), \( A_i | TC_i \) describes the angle or the type of curvature at \( P_i \), \( L_i \) describes the compared length of the edges connected at \( P_i \), and finally, \( C_i \) describes the convexity at \( P_i \).

In this study, the qualitative shape representation model (QSDM) of spatial objects is established on the basis of their research. The spatial objects are edge-connected, so we employ the other three features.

As shown in Figure 2, it is assumed that \( P_2, P_3, \) and \( P_4 \) are three points of the spatial object \( S \), which are connected in turn. At point \( P_3 \), a three-tuple is defined to represent the relevant point,

\[ \langle A_P, L_P, C_P \rangle \],

where \( A_P \) denotes the angle between line segments \( P_2P_3 \) and \( P_3P_4 \), \( L_P \) denotes the ratio of the length of the line segments, and \( C_P \) denotes the concavity and convexity of point \( P_3 \).

![Figure 2. Sketch of shape parameter computation.](image)

3.2.1. Angle Computation

The angle between two adjacent line segments of the polygon is computed at relevant points by using the inner product. For one-dimensional vectors \( a \) and \( b \) which are constructed by connecting a relevant point \( P_3 \) to its previous point \( P_2 \) and its next point \( P_4 \), respectively, the angle is computed by GA, which encodes the inner and the outer product; the inner product is proportional to the cosine, and the outer product is the sine. Thus, the value of the angle can be computed by the \( \text{arctan2} \) function. At first, we partition the angle into five intervals, \([0, 40], (40, 85], (85, 95], (95, 140], (140, 180] \), and the corresponding semantic expressions of the five intervals are \{very_acute, acute, right, obtuse, very_obtuse\}. In this work, we use an open ball \( B_r(c) \) (Borelian notation) where \( c = (a + b) / 2 \) (center) and \( r = (b - a) / 2 \) (radius). Given two internals, \( I_1 = (a_1, b_1) = B_{r_1}(c_1) \) and \( I_2 = (a_2, b_2) = B_{r_2}(c_2) \), a family of distances between intervals was defined [51], which depends on three parameters as follows:

\[ d^2(I_1, I_2) = (\Delta c \Delta r) A \left( \frac{\Delta c}{\Delta r} \right), \]
where $\Delta c = c_2 - c_1$, $\Delta r = r_2 - r_1$, and $A$ is a symmetrical $2 \times 2$ matrix of weights. Here, we use the most natural choice for the $A$ matrix, which is the identity matrix that provides the next distance:

$$d^2(I_1, I_2) = \sqrt{\Delta^2 c + \Delta^2 r} = \sqrt{(c_2 - c_1)^2 + (r_2 - r_1)^2}.$$  \hspace{1cm} (3)

Using Borelian notation and the interval distance, for example, for the two intervals $[0, 40]$ and $[40, 85]$, $I_1 = (0, 40) = B_{20}(20)$ and $I_2 = (40, 85) = B_{22.5}(62.5)$, $d^2(I_1, I_2) = \sqrt{(62.5 - 20)^2 + (22.5 - 20)^2} = 42.6$, which is the distance between the very acute and acute. Using the same process, we obtain all the angle relation distances, as shown in Table 1.

Table 1. Angle relation distances.

|         | Very Acute | Acute   | Right   | Obtuse  | Very Obtuse |
|---------|------------|---------|---------|---------|-------------|
| Very acute | 0.0        | 42.6    | 71.6    | 97.5    | 140.0       |
| Acute   | 42.6       | 0.0     | 32.6    | 55.0    | 97.5        |
| Right   | 71.6       | 32.6    | 0.0     | 32.6    | 71.6        |
| Obtuse  | 97.5       | 55.0    | 32.6    | 0.0     | 42.6        |
| Very obtuse | 140.0 | 97.5    | 71.6    | 42.6    | 0.0         |

3.2.2. Ratio of Length Computation

The lengths of two adjacent line segments of polygon feature points are computed using the inner product. The Euclidean distance between two points is directly embedded in the inner product computation with the expression $p_1 \cdot p_2 = -\frac{1}{2}d^2_E(p_1, p_2)$, and the ratio of the lengths is the ratio of the lengths of two adjacent line segments.

We partition the ratio of the lengths into seven intervals, $\{(0, 0.4], (0.4, 0.6], (0.6, 0.9], (0.9, 1.1], (1.1, 1.9], (1.9, 2.1], (2.1, 10]\}$, and the corresponding semantic expressions of the seven intervals are \{much_shorter (msh), half_length (hl), a_bit_shorter (absh), similar_length (sl), a_bit_longer (abl), double_length (dl), much_longer (ml)\}. Using Borelian notation and the interval distance, the computing method is the same as Formula (3), and we obtain the angle relation distances, as shown in Table 2.

Table 2. Length ratio relation distances.

| msh       | hl       | absh     | sl       | abl     | dl       | ml       |
|-----------|----------|----------|----------|---------|----------|----------|
| msh       | 0.00     | 0.32     | 0.55     | 0.81    | 1.32     | 1.80     | 6.95     |
| hl        | 0.32     | 0.00     | 0.25     | 0.50    | 1.04     | 1.50     | 6.75     |
| absh      | 0.55     | 0.25     | 0.00     | 0.25    | 0.79     | 1.25     | 6.52     |
| sl        | 0.81     | 0.50     | 0.25     | 0.00    | 0.58     | 1.00     | 6.35     |
| abl       | 1.32     | 1.04     | 0.79     | 0.58    | 0.00     | 0.58     | 5.77     |
| dl        | 1.80     | 1.50     | 1.25     | 1.00    | 0.58     | 0.00     | 5.59     |
| ml        | 6.95     | 6.75     | 6.52     | 6.35    | 5.77     | 5.59     | 0.00     |

3.2.3. Concavity and Convexity Judgment

The concavity and convexity of the angle between two adjacent line segments of the polygon are judged using the inner product. For example, for point $P_2$, its concavity and convexity can be determined by the relation between a one-dimensional vector $a = P_1P_2$ and the point $P_3$. If the point $P_3$ is on the left of vector $a$, the point $P_2$ is concave; otherwise, $P_2$ is convex. The concrete computation can construct a one-dimensional vector $c = P_1P_3$ and a one-dimensional vector $m$ that is perpendicular to $P_1P_2$ on the right, and then the sign of the inner product $c \cdot m$ is examined. If $c \cdot m > 0$, point $P_3$ is on the right of vector $a$. If $c \cdot m < 0$, point $P_3$ is on the left of vector $a$. For concavity relation distance, it equals 0 if they are the same; otherwise, it equals 1. For two concave or convex relations, the distance is 0, and for one concave and the other convex, the distance is 1.

The computations of the three parameters above are the components of the shape computation, which mainly rely on inner product computation. An inner product contains...
the information of the angle and distance, which can obtain the three-tuple quantitative values of the shape expression model concisely and intuitively and lay the foundation for the computation of the shape similarity.

For two objects, we adopt the following formula to compute the shape similarity between them:

\[
Sim(RP_A, RP_B) = 1 - \sum_{i \in \{A, L, C\}} w_i \frac{ds(i)}{Ds(i)},
\]

(4)

where \(RP_A\) and \(RP_B\) denote the relevant points of objects \(A\) and \(B\), respectively; \(ds(i)\) denotes the distance between two relations; \(Ds(i)\) denotes the maximum distance; and \(w_i\) denotes weights for each relation.

### 3.3. Topological Relations Similarity Computation

Zong [47] performed research on spatial topological relations based on geometric algebra and discussed the topological relations between a point, line segment, and triangle. Chen [48] sought the topological relations between two polygons by decomposing a polygon into triangles. In this article, we do not distinguish the topological relations as Chen did, but rather, we take the polygon as a whole.

The spatial topological relations are determined based on the research of Zong and Chen. We divide the object into the interior, boundary, and exterior (the point can be divided into the interior and the exterior, without the boundary), use the geometric product, and construct different discriminants. Thus, topological relations between objects of different types can be obtained.

#### 3.3.1. Point and Other Objects

(1) Topological relations between two points

For two points \(A\) and \(B\), the discriminant is constructed as follows:

\[
t = A \land B.
\]

(5)

If \(t = 0\), \(A\) intersects \(B\), and the topological relation is pp-equal. If \(t <> 0\), \(A\) does not intersect \(B\), and the topological relation is pp-disjoint. The topological relations are shown in Figure 3.

![Figure 3. Topological relations between two points.](image)

Using the topological relation operation between two points, we obtain two relations: pp-disjoint and pp-equal. Figure 4 shows the corresponding conceptual neighborhood diagram. For two points, when the topological relation is the same as the other two, the relation distance is 0; otherwise, the relation distance is 1.

![Figure 4. Conceptual neighborhood diagram of topological relations between two points.](image)
(2) Topological relations between point and line segment

A line segment is a straight line with a boundary constraint. For a straight line \( L \) passing through points \( A \) and \( B \), the judgment expression is \( L = A \hat{\wedge} B \hat{\wedge} e_i \). The topological relations between point \( P \) and line segment \( AB \) can be determined by the two following expressions:

\[
t_1 = (A \wedge P \wedge e_i) L, \quad t_2 = (P \wedge B \wedge e_i) L. \tag{6}
\]

If \( t_1 > 0 \) and \( t_2 > 0 \), \( P \) is within \( AB \), and the topological relation between them is pl-inside. If \( t_1 = 0 \) or \( t_2 = 0 \), \( P \) coincides with the edges of \( AB \), and the topological relation between them is pl-touch. If \( t_1 < 0 \) or \( t_2 < 0 \), \( P \) does not intersect with \( AB \), and the topological relation between them is pl-disjoint. The topological relations are shown in Figure 5.

![Figure 5. Topological relations between a point and a line segment.](image)

Using the topological relation operation between a point and a line segment, we obtain three relations: pl-disjoint, pl-touch, and pl-inside. Figure 6 shows the corresponding conceptual neighborhood diagram. The topological relation distances which are decided according to the conceptual neighborhood diagram are shown in Table 3.

![Figure 6. Conceptual neighborhood diagram of topological relations between a point and a line segment.](image)

| Point–Line | pl-Disjoint | pl-Touch | pl-Inside |
|------------|-------------|----------|-----------|
| pl-disjoint| 0           | 1        | 2         |
| pl-touch   | 1           | 0        | 1         |
| pl-inside  | 2           | 1        | 0         |

(3) Topological relations between point and polygon

The relations can be computed using the following judgment expression:

\[
t = (P \wedge L) \Phi,
\]

where \( P \) is the point, \( L \) is the boundary of the polygon, and \( \Phi \) is the plane that contains the polygon \( M. \Phi = C \wedge e_i \), \( C \) is the boundary circle, \( e_i \) denotes infinity, and \( L \) denotes the straight line at which the boundary of the polygon \( M \) is located.

If \( t < 0 \) for all judgment expressions, \( P \) is inside \( M \). The topological relation between them is ps-inside. If \( t = 0 \), \( P \) is in the boundary of \( M \). The topological relation between them is ps-touch. In other situations, \( P \) is outside \( M \). The topological relation between them is ps-disjoint. The topological relations are shown in Figure 7.
Using the topological relation operation between a point and a polygon, we obtain three relations: ps-disjoint, ps-touch, and ps-inside. Figure 8 shows the corresponding conceptual neighborhood diagram. The topological relation distances which are decided according to the conceptual neighborhood diagram are shown in Table 4.

| Point–Polygon | ps-Disjoint | ps-Touch | ps-Inside |
|---------------|-------------|----------|-----------|
| ps-disjoint   | 0           | 1        | 2         |
| ps-touch      | 1           | 0        | 1         |
| ps-inside     | 2           | 1        | 0         |

3.3.2. Two Line Segments

Using the operation between a point and a line segment, we can construct a computing strategy to determine the topological relations between two line segments. For two line segments $L_1$ and $L_2$, the relation between the two edge points of $L_1$ and the line segment $L_2$ is computed. When one of the edge points of $L_1$ is within $L_2$ or coincident with the edge points of $L_2$, the topological relation is ll-touch. When the two edges of $L_1$ coincide with two edges of $L_2$, the topological relation is ll-equal. When the two edge points of $L_1$ are inside $L_2$ and the two edge points of $L_2$ are outside $L_1$, the topological relation is ll-contains. When the interior of $L_2$ intersects with that of $L_1$, the topological relation is ll-overlap. When both the interior and the two edge points of $L_2$ do not intersect with the interior and the two edge points of $L_1$, the topological relation is ll-disjoint. The topological relations are shown in Figure 9.
Using the topological relation operation between two line segments, we obtain six relations: ll-disjoint, ll-touch, ll-overlap, ll-equal, ll-contains, and ll-inside. Figure 10 shows the corresponding conceptual neighborhood diagram. The topological relation distances which are decided according to the conceptual neighborhood diagram are shown in Table 5.

Table 5. Topological relation distances between two line segments.

| Line–Line | ll-disjoint | ll-touch | ll-overlap | ll-equal | ll-contains | ll-inside |
|-----------|-------------|----------|------------|----------|-------------|----------|
| ll-disjoint | 0 | 1 | 2 | 3 | 4 | 4 |
| ll-touch | 1 | 0 | 1 | 2 | 3 | 3 |
| ll-overlap | 2 | 1 | 0 | 1 | 2 | 2 |
| ll-equal | 3 | 2 | 1 | 0 | 1 | 1 |
| ll-contains | 4 | 3 | 2 | 1 | 0 | 2 |
| ll-inside | 4 | 3 | 2 | 1 | 2 | 0 |
3.3.3. Line Segment and Polygon

Using the operation between a point and a polygon and the operation between two line segments, we compute the topological relations as follows. For the two edge points of $L$, if both are inside polygon $S$, the topological relation between $L$ and $S$ is ls-inside1. If one edge point of $L$ is inside polygon $S$ and the other edge point is on the boundary of $S$, the topological relation between $L$ and $S$ is ls-inside2. If the two edge points of $L$ are on the boundary of $S$, the topological relation between $L$ and $S$ is ls-inside3. If one edge point of $L$ is on the boundary of $S$ and the other is outside $S$, the topological relation between $L$ and $S$ is ls-touch. If one edge point of $L$ is inside $S$ and the other is outside $S$, the topological relation between $L$ and $S$ is ls-overlap1.

If the two edge points of $L$ are outside $S$, the relation between the line segment $L$ and the boundary of $S$ is judged. If $L$ does not intersect with the boundary of $S$, the topological relation between $L$ and $S$ is ls-disjoint. If $L$ intersects with the boundary of $S$, the topological relation between $L$ and $S$ is ls-overlap2.

In summary, we can distinguish four topological relations: ls-disjoint, ls-overlap, ls-inside, and ls-touch. The relations ls-inside1, ls-inside2, and ls-inside3 can be recognized as ls-inside, and the relations ls-overlap1 and ls-overlap2 can be recognized as ls-overlap. The topological relations are shown in Figure 11.

![Figure 11. Topological relations between a line segment and a polygon.](image)

Using the topological relation operation between a line segment and a polygon, we obtain four relations: ls-disjoint, ls-touch, ls-overlap, and ls-inside. Figure 12 shows the corresponding conceptual neighborhood diagram. The topological relation distances which are decided according to the conceptual neighborhood diagram are shown in Table 6.

![Figure 12. Conceptual neighborhood diagram of topological relations between a line segment and a polygon.](image)
Table 6. Topological relation distance between a line segment and a polygon.

| Line–Polygon | ls-disjoint | ls-touch | ls-overlap | ls-inside |
|--------------|-------------|----------|------------|-----------|
| ls-disjoint  | 0           | 1        | 2          | 3         |
| ls-touch     | 1           | 0        | 1          | 2         |
| ls-overlap   | 2           | 1        | 0          | 1         |
| ls-inside    | 3           | 2        | 1          | 0         |

3.3.4. Two Polygons

Based on the operation between a point and a line segment, between a point and a polygon, and between two line segments, we compute the topological relations as follows.

For all vertices \( U_1, U_2, \ldots, U_m \) of \( S_1 \), if all are inside \( S_2 \), the operation between a point and a polygon is used, and the condition is recorded as \( C_1 \). If all the vertices are outside \( S_2 \), the condition is recorded as \( C_2 \). If they are all on the boundary of \( S_2 \), the condition is recorded as \( C_3 \). If some of the vertices are inside \( S_2 \) and some are outside \( S_2 \), the condition is recorded as \( C_4 \). If some of the vertices are inside \( S_2 \) and some are on the boundary of \( S_2 \), the condition is recorded as \( C_5 \). If some of the vertices are outside \( S_2 \) and some are on the boundary of \( S_2 \), the condition is recorded as \( C_6 \). Likewise, we obtain the conditions \( D_1, D_2, D_3, D_4, D_5, \) and \( D_6 \).

If \( C_1 \) is satisfied but \( D_1 \) is not, the topological relation between \( S_1 \) and \( S_2 \) is ss-contains. If \( D_1 \) is satisfied but \( C_1 \) is not, the topological relation between \( S_1 \) and \( S_2 \) is ss-inside. If both \( C_2 \) and \( D_2 \) are satisfied, the topological relation between \( S_1 \) and \( S_2 \) is ss-disjoint. If both \( C_3 \) and \( D_3 \) are satisfied, the topological relation between \( S_1 \) and \( S_2 \) is ss-equal. If \( C_4 \) or \( D_4 \) is satisfied or both are, the topological relation between \( S_1 \) and \( S_2 \) is ss-overlap. If \( D_5 \) is satisfied, the topological relation between \( S_1 \) and \( S_2 \) is ss-covers. If \( C_5 \) is satisfied, the topological relation between \( S_1 \) and \( S_2 \) is ss-covered-by. If both \( C_6 \) and \( D_6 \) are satisfied, the topological relation between \( S_1 \) and \( S_2 \) is ss-touch. The topological relations are shown in Figure 13.

![Figure 13. Topological relations between two polygons.](image)
Using the topological relation operation between two polygons, we obtain eight relations: ss-disjoint, ss-touch, ss-overlap, ss-covers, ss-covered-by, ss-contains, ss-inside, and ss-equal. Figure 14 shows the corresponding conceptual neighborhood diagram. The topological relation distances which are decided according to the conceptual neighborhood diagram are shown in Table 7. The topological relations between two polygons can be distinguished using the same topological relations with the RCC8 model.

![Conceptual neighborhood diagram of topological relations between two polygons.](image)

**Table 7.** Topological relation distances between a line segment and a polygon.

| Polygon–Polygon | ss-Disjoint | ss-Touch | ss-Overlap | ss-Covers | ss-Covered-by | ss-Contains | ss-Inside | ss-Equal |
|------------------|-------------|----------|------------|-----------|--------------|-------------|-----------|----------|
| ss-disjoint      | 0           | 1        | 2          | 3         | 3            | 4           | 4         | 3        |
| ss-touch         | 1           | 0        | 1          | 2         | 2            | 3           | 3         | 2        |
| ss-overlap       | 2           | 1        | 0          | 1         | 1            | 2           | 2         | 1        |
| ss-covers        | 3           | 2        | 1          | 0         | 2            | 1           | 2         | 1        |
| ss-covered-by    | 3           | 2        | 1          | 2         | 0            | 2           | 1         | 1        |
| ss-contains      | 4           | 3        | 2          | 1         | 2            | 0           | 2         | 1        |
| ss-inside        | 4           | 3        | 2          | 2         | 1            | 2           | 0         | 1        |
| ss-equal         | 3           | 2        | 1          | 1         | 1            | 1           | 1         | 0        |

3.3.5. Topological Relation Similarity Computation Strategy

$m$ spatial objects in query scene $V$ are denoted as $v_1, v_2, \ldots, v_m$. $n$ spatial objects in the query database $U$ are denoted as $u_1, u_2, \ldots, u_n$. We adopt the following strategy to compute the topological relations between two scenes. First, the topological relation operation is used to resolve the relation between two objects in the query scene, which is denoted as $T_1$. Furthermore, we obtain the relation between two objects in the database scene, which is denoted as $T_2$. By utilizing the corresponding conceptual neighborhood diagram, the relation distance is computed. Furthermore, the similarity $S_{Ri}$ between $T_1$ and $T_2$ is computed.

We compute all topological relation similarities between objects in the query scene and corresponding objects in the database scene as follows:

$$S_{Rel} = \frac{\sum_{i=1}^{t} \alpha(t-1)/2 W_{Ri} \cdot S_{Ri}}{\sum_{i=1}^{t} \alpha(t-1)/2 W_{Ri}}$$

(8)
where \( W_{Ri} \) denotes the weight values of the topological relations. In this paper, the weight values are the same; however, for future work, the weight values can be different according to the actual scene. For example, the topological relation inside may be more important than disjoint. \( S_{Ri} \) denotes the similarity between two relations, and \( t \) denotes the number of matched objects.

3.4. Direction Relation Similarity Computation

The outer product realizes the dimensional and geometric operations, while the inner product provides the angle and distance information. For two geometric objects \( O_1 \) and \( O_2 \) of arbitrary dimensions, the metric information between two basic objects is as follows:

\[
\cos(\theta) = \frac{O_1 \cdot O_2}{|O_1||O_2|},
\]

\[
\theta = \angle(O_1, O_2) = \arccos\left(\frac{O_1 \cdot O_2}{|O_1||O_2|}\right).
\]

For example, as shown in Figure 15a, for two vectors \( O_1 \) and \( O_2 \), the inner product is the distance that \( O_1 \) projects onto \( O_2 \). In Figure 15b, for the vector \( O_1 \) and the plane \( O_2 \), the inner product is the vector that is in \( O_2 \) and perpendicular to \( O_1 \). Owing to high-dimensional scalability, the operation is conferrable to arbitrary dimensions, not limited to vectors [52].

![Figure 15. Inner product. (a) Inner product between two vectors; (b) Inner product between one vector and one plane.](image)

If \( P_1(0,1) \) and \( P_2(1,0) \) belong to a 2D Euclidean space, by converting them into the conformal space, we obtain \( C_{ga \_P_1} = e_2 + 0.5e_\infty + e_0 \) and \( C_{ga \_P_2} = e_1 + 0.5e_\infty + e_0 \). The origin point is \( C_{ga \_O} = e_0 \). The outer product between two points and infinity can be used to construct a line. Thus, the line \( P_1P_2 \) can be represented as \( C_{ga \_P_1} \wedge C_{ga \_P_2} \wedge e_\infty \). The magnitude of \( P_1P_2 \) is \( |P_1P_2| = \sqrt{2} \), and the magnitude of the unit base vector is \( |OE| = 1 \). Thus, the inner product \( P_1P_2 \cdot OE = |P_1P_2| |OE| \cos \theta \), and the computation result is \( \cos \theta = -\sqrt{2}/2 \), that is, \( \theta = 135^\circ \).

The conformal space is angle-preserving. The angle computed by the inner product in conformal space reflects the angle in Euclidean space. The cosine function was symmetric within the scope of \([0^\circ, 360^\circ]\). If \( y_{OA} - y_{OB} < 0 \), then \( \theta \in [0, 180^\circ] \). If \( y_{OA} - y_{OB} > 0 \), then \( \theta \in [180^\circ, 360^\circ] \).

In CGA, according to the Grassmann hierarchical structure, the minimum bounding circle can be constructed by the vertices of the polygon. The radius of the circle is obtained by \( r = \sqrt{\frac{-c^2}{(C_\alpha e_f)}} \), and the center of the circle is computed by \( O = Ce_fC \).

Thus, for two geometric objects \( A \) and \( B \), the direction relation operation is implemented as follows. (1) The center points \( O_A \) and \( O_B \) of the minimum bounding circles are computed for \( A \) and \( B \), respectively. (2) The inner product between \( O_AO_B \) and the unit base vector are computed, and then the angle value can be computed.

The angle value can be mapped to the corresponding angle interval, and the corresponding semantic expression can be acquired. Figure 16 illustrates the relation between the direction relation semantic expression and the angle value.
The direction relation semantic expression is as follows: \{N, NE, E, SE, S, SW, W, NW\}, representing north, northeast, east, southeast, south, southwest, west, and northwest, respectively. The corresponding value intervals are \([67.5, 112.5), [22.5, 67.5), [337.5, 22.5), [292.5, 337.5), [247.5, 292.5), [202.5, 247.5), [157.5, 202.5), [112.5, 157.5]\). In Borelian notation, we obtain \{B_{22.5}(90), B_{22.5}(45), B_{22.5}(0), B_{22.5}(315), B_{22.5}(270), B_{22.5}(225), B_{22.5}(180), and B_{22.5}(135)\}. Using Formula (3), we find all \(r = 22.5\), and the distance \(d = 45\) between two adjacent directions. Hence, we simplify the distance, which is similar to the conceptual neighborhood diagram.

The direction relation is reflexive. When the direction relation distance surpasses 4, the format \(\text{Dir}_{\text{Max}} - \text{Dir}_{\text{Cal}}\) is used to compute the relation distance. The final direction relation distances are shown in Table 8.

### Table 8. Direction relation distance.

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

We employ the direction relation operation to compute the angle value between two objects, map the angle value to an angle interval, obtain the semantic expression, and compute the distance and similarity of the direction relations. The similarity computation formula of the direction relation is as follows:

\[
\text{Sim}(\text{Dr}_0, \text{Dr}_1) = 1 - \frac{dr}{Dr},
\]

where \(dr\) denotes the direction relation distance and \(Dr\) denotes the maximal distance.

### 3.5 Distance Relation Similarity Computation

As discussed in Section 3.4, the inner product contains the angle and distance information. In this section, we compute the distance information. The inner product is 
\[
C2ga_{P1} \cdot C2ga_{P2} = (e_2 + 0.5e_\infty + e_0)(e_1 + 0.5e_\infty + e_0).
\]
Based on the equations \(e_0^2 - e_\infty^2 = 0\) and \(e_0e_\infty = -1\), the value of the inner product is \(-1\). The Euclidean distance is embedded...
in the inner product based on the expression $a \cdot b = -\frac{1}{2}d_E^2(a, b)$, and thus, the Euclidean distance is $\sqrt{(-2) * (-1)} = \sqrt{2}$.

For two geometric objects $A$ and $B$, the distance relation operation is implemented as follows. (1) The center points $O_A$ and $O_B$ of the minimum bounding circles are computed for $A$ and $B$, respectively. (2) The inner product between $O_A$ and $O_B$ is computed, and then the distance value can be computed.

In CGA, the distance relation operation is used to compute the distance between two objects. We determine the relation between the quantitative distance value and the distance interval and obtain the semantic distance expression. Using the distance relation table, we can compute the distance relation similarity.

In an actual scene, the distance value range is uncertain. We use the ratio of the specific value and maximal value to represent the distance relation. In this study, we define the distance relation semantics expression as \{very close (vc), close (cl), a bit close (abc), commensurate (co), a bit far (abf), far (f), very far (vf)\}. The corresponding distance intervals are ([0, 0.1), [0.1, 0.16), [0.16, 0.2), [0.2, 0.3), [0.3, 0.5), [0.5, 0.7), [0.7, $\infty$}). In Borelian notation, we obtain $[B_{0.05}(0.05), B_{0.03}(0.13), B_{0.02}(0.18), B_{0.05}(0.25), B_{0.1}(0.4), B_{1.1}(0.6), B_{0.15}(0.85)]$. The distance relation distances computed by using Formula (3) are shown as Table 9.

### Table 9. Distance relation distance.

| Length | vc  | cl  | abc | co  | abf | f   | vf  |
|--------|-----|-----|-----|-----|-----|-----|-----|
| vc     | 0.000 | 0.082 | 0.133 | 0.200 | 0.354 | 0.552 | 0.806 |
| cl     | 0.082 | 0.000 | 0.051 | 0.122 | 0.279 | 0.475 | 0.730 |
| abc    | 0.133 | 0.051 | 0.000 | 0.076 | 0.234 | 0.428 | 0.682 |
| co     | 0.200 | 0.122 | 0.076 | 0.000 | 0.158 | 0.354 | 0.608 |
| abf    | 0.354 | 0.279 | 0.234 | 0.158 | 0.000 | 0.200 | 0.453 |
| f      | 0.552 | 0.475 | 0.428 | 0.354 | 0.200 | 0.000 | 0.255 |
| vf     | 0.806 | 0.730 | 0.682 | 0.608 | 0.453 | 0.255 | 0.000 |

The similarity computation formula for the distance relation is as follows:

$$Sim(Dis_0, Dis_1) = 1 - \frac{ds}{Ds},$$

where $ds$ denotes the relation distance and $Ds$ denotes the maximal distance.

### 3.6. Spatial Geometric Similarity Computation

Based on the whole process shown in Figure 1, we designed a retrieval method taking the shape and spatial relations (including topological, direction, and distance relation) into account. During the whole process, we assume that the weights are equivalent for the objects and the spatial relations between them. The topology, direction, and distance relations have the same effect on the spatial relation.

The scenes in Figure 17 are considered as an example. We apply the object shape and spatial relation similarity computation model to analyze the example. For the query scene (a), using the multivector and corresponding relation operations, we obtain the following integrating representation [46] form:

$$GS_a = \{\langle A, B \rangle + \langle A, C \rangle + \langle A, D \rangle + \langle A, E \rangle + \langle B, C \rangle + \langle B, D \rangle + \langle B, E \rangle + \langle C, D \rangle + \langle C, E \rangle + \langle D, E \rangle\}$$

The shape and spatial relation information are implied in the representation. The related operations including topological relation, direction relation, and distance relation can be called to compute the corresponding relations directly. The topological, direction, and distance relations in the query scene are listed in Tables 10–12, respectively.
Figure 17. (a) Query scene. (b–e) Database scenes.

Table 10. Topological relations in the query scene (a) in Figure 17.

|     | A₁ | B₁ | C₁ | D₁ | E₁ |
|-----|----|----|----|----|----|
| A₁  | ss-equal | ss-touch | ss-disjoint | ss-disjoint | ss-disjoint |
| B₁  | ss-touch  | ss-equal | ss-disjoint | ss-disjoint | ss-disjoint |
| C₁  | ss-disjoint | ss-disjoint | ss-equal  | ss-disjoint | ss-disjoint |
| D₁  | ss-disjoint | ss-disjoint | ss-disjoint | ss-equal  | ss-touch  |
| E₁  | ss-disjoint | ss-disjoint | ss-disjoint | ss-touch  | ss-equal  |

Table 11. Direction relations in the query scene (a) in Figure 17.

|     | A₁ | B₁ | C₁ | D₁ | E₁ |
|-----|----|----|----|----|----|
| A₁  | O  | NW | W  | SW | W  |
| B₁  | SE | O  | W  | SW | SW |
| C₁  | E  | E  | O  | SE | S  |
| D₁  | NE | NE | NW | O  | W  |
| E₁  | E  | NE | N  | E  | O  |

Table 12. Distance relations in the query scene (a) in Figure 17.

|     | A₁ | B₁ | C₁ | D₁ | E₁ |
|-----|----|----|----|----|----|
| A₁  | /  | abf| vf | vf | vf |
| B₁  | abf| /  | vf | f  | vf |
| C₁  | vf | vf | /  | f  | f  |
| D₁  | vf | f  | f  | /  | co |
| E₁  | vf | vf | f  | co | /  |

For the database scene (b), following the same computation strategy, we obtain the topological, direction, and distance relations, as shown in Tables 13–15, respectively.
Table 13. Topological relations in the database scene (b) in Figure 17.

|      | A₂ | B₂ | C₂ | D₂ | E₂ |
|------|----|----|----|----|----|
| A₂   | ss-equal | ss-touch | ss-disjoint | ss-disjoint | ss-disjoint |
| B₂   | ss-touch | ss-equal | ss-disjoint | ss-disjoint | ss-disjoint |
| C₂   | ss-disjoint | ss-disjoint | ss-equal | ss-disjoint | ss-disjoint |
| D₂   | ss-disjoint | ss-disjoint | ss-disjoint | ss-equal | ss-disjoint |
| E₂   | ss-disjoint | ss-disjoint | ss-disjoint | ss-disjoint | ss-equal |

Table 14. Direction relations in the database scene (b) in Figure 17.

|      | A₂ | B₂ | C₂ | D₂ | E₂ |
|------|----|----|----|----|----|
| A₂   | O  | NW | W  | SW | W  |
| B₂   | SE | O  | W  | SW | SW |
| C₂   | E  | E  | O  | SE | S  |
| D₂   | NE | NE | NW | O  | W  |
| E₂   | E  | NE | N  | E  | O  |

Table 15. Distance relations in the database scene (b) in Figure 17.

|      | A₂  | B₂  | C₂  | D₂  | E₂  |
|------|-----|-----|-----|-----|-----|
| A₂   | /   | abf | vf  | f   | vf  |
| B₂   | abf | /   | vf  | f   | vf  |
| C₂   | vf  | /   | vf  | f   | vf  |
| D₂   | f   | f   | vf  | /   | abf |
| E₂   | vf  | vf  | f   | abf | /   |

Based on the spatial relations computed by the corresponding operations, the topological, direction, and distance relation similarities between (a) and (b) are $S_{Top} = (0.75 \times 1 + 1 \times 9)/10 = 0.975$, $S_{Dir} = (1 \times 10)/10 = 1.0$, and $S_{Dis} = ((1 - 0.255/0.806) \times 2 + (1 - 0.158/0.806) + 7 \times 1.0)/10 = 0.917$, respectively. For (a) and (b), the attribute and shape are the same, so the spatial similarity is $Sim(a, b) = 1/2 + (S_{Top} + S_{Dir} + S_{Dis})/6 = 0.968$.

In the database scene (c), the different colors represent different attributes, and only object B was different. When taking the shape and spatial relations into consideration, the similarities between (c) and (a) are equal, and both are 1. For the database scene (d), we obtain the spatial relation tables shown in Tables 16–18.

Table 16. Topological relations in the database scene (d) in Figure 17.

|      | A₄ | B₄ | C₄ | D₄ | E₄ |
|------|----|----|----|----|----|
| A₄   | ss-equal | ss-touch | ss-disjoint | ss-disjoint | ss-disjoint |
| B₄   | ss-touch | ss-equal | ss-disjoint | ss-disjoint | ss-disjoint |
| C₄   | ss-disjoint | ss-disjoint | ss-equal | ss-disjoint | ss-disjoint |
| D₄   | ss-disjoint | ss-disjoint | ss-disjoint | ss-equal | ss-disjoint |
| E₄   | ss-disjoint | ss-disjoint | ss-disjoint | ss-equal | ss-touch |

Table 17. Direction relations in the database scene (d) in Figure 17.

|      | A₄ | B₄ | C₄ | D₄ | E₄ |
|------|----|----|----|----|----|
| A₄   | O  | NW | W  | W  | W  |
| B₄   | SE | O  | NE | W  | W  |
| C₄   | E  | SW | O  | N  | NW |
| D₄   | E  | E  | S  | O  | W  |
| E₄   | E  | E  | SE | E  | O  |
Table 18. Distance relations in the database scene (d) in Figure 17.

|   | A_4 | B_4 | C_4 | D_4 | E_4 |
|---|-----|-----|-----|-----|-----|
| A_4 | / | abf | vf | vf | vf |
| B_4 | abf | / | vf | abf | f |
| C_4 | vf | vf | / | abf | f |
| D_4 | vf | abf | abf | / | abf |
| E_4 | vf | f | f | abf | / |

Based on the spatial relations computed by corresponding operations, the topological, direction, and distance relation similarities between (a) and (d) are $S_{Top} = (1 \times 10)/10 = 1.0$, $S_{Dir} = ((1 - 0.25) \times 3 + (1 - 0.75) \times 3 + 1 \times 4)/10 = 0.7$, and $S_{Dis} = ((1 - 0.2/0.806) \times 2 + (1 - 0.255/0.806) \times 1 + (1 - 0.158/0.806) \times 1 + 1.0 \times 6)/10 = 0.899$, respectively. The attributes and shapes of the two scenes are the same, so the shape similarity is 1. Hence, the spatial similarity is $Sim(a, d) = 1/2 + (S_{Top} + S_{Dir} + S_{Dis})/6 = 0.933$. According to the computation results, the spatial similarity between (b) and (a) is bigger than that between (d) and (a), so the scene (b) is more similar than (d) to (a).

For the database scene (e), the spatial relations are the same as in the query scene (a). The shape of the object C is different: it is a triangle in (a) and a square in (e). The shape similarity operation is used first to construct a three-tuple representation for the relevant points. For scene (a), the relevant point representations of object $C_1$ are <acute, sl, convex>, <acute, sl, convex>, and <acute, sl, convex>. Similarly, the point representations of object $C_5$ are <right, sl, convex>, <right, sl, convex>, <right, sl, convex>, and <right, sl, convex>. Thus, the shape similarity between $C_1$ and $C_5$ is $Sim = (1 - 32.6/(140.0 \times 3)) \times 3)/4 = 0.69$. Thus, the spatial similarity between scenes (a) and (e) is $S_{ae} = 0.5 + ((0.69 + 4)/5)/2 = 0.969$. The similarity values between each scene and the query scene are shown in Table 19.

Table 19. Geometric similarity values between each scene and query scene.

|         | $S_a$-$S_b$ | $S_a$-$S_c$ | $S_a$-$S_d$ | $S_a$-$S_e$ |
|---------|-------------|-------------|-------------|-------------|
| Similarity Value | 0.968 | 1.000 | 0.933 | 0.969 |

4. Case Study

4.1. Related Background

For resident areas in a city, public transportation stations are common facilities along the roads. Public transportation stations are simplified as point objects, roads as polyline objects, and resident buildings as polygons. The resident areas are spatial geometric scenes with objects of different types. The traditional methods cannot satisfy the retrieval demands for these scenes very well. In this section, we attempt to obtain a unified representation and computation for spatial geometric scenes to overcome the disunity of the representation and complexity of the spatial relation computation.

4.2. Retrieval Process

(1) An integrating representation model of geographical objects and geographical scenes is developed based on CGA. The model contains the geometric and spatial relation information for the scene. We can call related operations to resolve the information. For example, one query scene is shown in Figure 18. The database scene is shown in Figure 19.
(2) Objects that satisfy the specified similarity threshold with the scene’s central objects are resolved. Three parameters, i.e., closeness, perimeter, and area, are employed to compute the central objects. The closeness computation formula is as follows:

$$FormFactor = \frac{4\pi \times Area_{Object}}{Perimeter_{Object}^2}. \quad (14)$$

The normalization formula is as follows:

$$Val = \frac{Com_A - Com_B}{Max - Min}. \quad (15)$$

Reasonable weights for each normalized parameter are assigned, and their weighted sum is calculated as follows:

$$Val_{final} = \sum_{i=1}^{n} w_i Val_i. \quad (16)$$

(3) The Voronoi graph and corresponding sub-scenes based on the neighborhood relations are constructed.

(4) The spatial geometric similarity computation method is used, and then the similarity between each sub-scene and the query scene is obtained. The retrieval results are shown in Figure 20.
Figure 20. Retrieval results.

The method proposed in this study can be used to retrieve spatial scenes with objects of different types. The retrieval results will be helpful for real estate developers and civilians to retrieve similar spatial geometric scenes. It is of great scientific significance and practical value to conduct spatial scene retrieval.

5. Conclusions

The amount of spatial geometric data in urban scenarios is increasing. In this study, we propose a method of computing the spatial geometric similarity in urban spatial patterns based on CGA. Operations for the shape and spatial relations are constructed using the inner, outer, and geometric products in CGA, and the relation distance tables are developed by means of conceptual neighborhood diagrams and interval distances. Then, the process of spatial geometric similarity computation is designed and realized based on the expression and computation of the object shape and spatial relations between two objects. From the aspects of the spatial object shape, spatial topology, direction, and distance between two objects, the spatial geometric similarity is evaluated more completely, which is important in the field of spatial data retrieval. In further research, on the one hand, semantic similarity should be taken into consideration for spatial similarity computation. The relations between the topology, direction, and distance can be deduced before the spatial similarity computation to improve the retrieval efficiency. On the other hand, we will adopt some computation strategies, such as parallel computing using MapReduce, to improve the efficiency. The method proposed in this paper can carry out comprehensive quantitative calculation by integrating spatial topology, direction, distance, and spatial object, which has important reference significance for urban planning and city site selection, providing a new perspective for spatial planning.

Author Contributions: Writing—original draft, X.J.; Writing—review & editing, Y.H. and F.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China grant number 41471313.

Data Availability Statement: The data is not available due to legal restrictions. The authors of this study did not agree for their data to be shared publicly, so supporting data is not available.

Conflicts of Interest: The authors declare no conflict of interest.
31. Doran, C.; Gullans, S.R.; Lasenby, A.; Lasenby, J.; Fitzgerald, W. Geometric Algebra for Physicists; Cambridge University Press: Cambridge, UK, 2003.
32. Pavšič, M. Clifford space as a generalization of spacetime: Prospects for QFT of Point Particles and Strings. Found. Phys. 2005, 35, 1617–1642. [CrossRef]
33. Cameron, J.; Lasenby, J. Oriented conformal geometric algebra. Adv. Appl. Clifford Algebras 2008, 18, 523–538. [CrossRef]
34. Hughes, J.F. Oriented Projective Geometry: A Framework for Geometric Computations (Jorge Stolfi). SIAM Rev. 1993, 35, 321–323. [CrossRef]
35. Fuchs, L.; Théry, L. Implementing geometric algebra products with binary trees. Adv. Appl. Clifford Algebras 2014, 24, 589–611. [CrossRef]
36. Yuan, L.; Lü, G.; Luo, W.; Yu, Z.; Yi, L.; Sheng, Y. Geometric algebra method for multidimensionally-unified GIS computation. Chin. Sci. Bull. 2012, 57, 802–811. [CrossRef]
37. Yuan, L.; Yu, Z.; Chen, S.; Luo, W.; Wang, Y.; Lü, G. CAUSTA: Clifford algebra-based unified spatio-temporal analysis. Trans. GIS 2010, 14, 59–83. [CrossRef]
38. Yuan, L.; Yu, Z.; Luo, W.; Hu, Y.; Feng, L.; Zhu, A.-X. A hierarchical tensor-based approach to compressing, updating and querying geospatial data. IEEE Trans. Knowl. Data Eng. 2014, 27, 312–325. [CrossRef]
39. Yuan, L.; Yu, Z.; Luo, W.; Yi, L.; Hu, Y. Pattern forced geophysical vector field segmentation based on Clifford FFT. Comput. Geosci. 2013, 60, 63–69. [CrossRef]
40. Yuan, L.; Yu, Z.; Luo, W.; Yi, L.; Lü, G. Geometric algebra for multidimension-unified geographical information system. Adv. Appl. Clifford Algebras 2013, 23, 497–518. [CrossRef]
41. Yuan, L.; Yu, Z.; Luo, W.; Yi, L.; Lü, G. Multidimensional-unified topological relations computation: A hierarchical geometric algebra-based approach. Int. J. Geogr. Inf. Sci. 2014, 28, 2435–2455. [CrossRef]
42. Yuan, L.; Yu, Z.; Luo, W.; Zhang, J.; Hu, Y. Clifford algebra method for network expression, computation, and algorithm construction. Math. Methods Appl. Sci. 2014, 37, 1428–1435. [CrossRef]
43. LÜ, G.; YUAN, L.; YU, Z.; LUO, W.; ZHOU, L.; YAN, Z. Role of Geometric Algebra in High Dimensional Space Representation of Geographic Information. Geomat. Inf. Sci. Wuhan Univ. 2020, 45, 1328–1333.
44. Li, S. Research on Formal Expression and Parallel Computation of Spatial Relations Based on Conformal Geometric Algebra; Nanjing Normal University: Nanjing, China, 2021.
45. Zhang, J.-Y.; Yin, P.-C.; Li, G.; Shi, Z.-F.; Liu, Y.-Z.; Yu, J.-Y. Study on Construction for the Three-Dimensional Cadastral Data Model Based on Conformal Geometric Algebra. Geogr. Geo-Inf. Sci. 2018, 5, 20. [CrossRef]
46. Zhang, F.; Jiang, X.; Zhang, X.; Wang, Y.; Du, Z.; Liu, R. Unified spatial intersection algorithms based on conformal geometric algebra. Math. Probl. Eng. 2016, 2016. [CrossRef]
47. Zong, Z. Spatial Relationship Expression and Computation Model Based on the Geometric Algebra; Nanjing Normal University: Nanjing, China, 2013.
48. Chen, M. Study on Spatio-Temporal Topological Relationship Expression and Computation Based on Conformal Geometric Algebra; Zhejiang University: Hangzhou, China, 2014.
49. Hongbo, L. Conformal geometric algebra-a new framework for computational geometry. J. Comput. Aided Des. Comput. Graph. 2005, 17, 2383.
50. Falomir, Z.; Gonzalez-Abril, L.; Museros, L.; Ortega, J.A. Measures of similarity between objects based on qualitative shape descriptions. Spat. Cogn. Comput. 2013, 13, 181–218. [CrossRef]
51. Gonzalez Abril, L.; Velasco Morente, F.; Angulo, C.; Ortega Ramirez, J.A.; Ruiz Vegas, F. Sobre núcleos, distancias y similitudes entre intervalos. Int. Artif. 2004, 8, 111–117. [CrossRef]
52. Hildenbrand, D.; Benger, W.; Zhaoyuan, Y. Analyzing the inner product of 2 circles with Gaalop. Math. Methods Appl. Sci. 2018, 41, 4049–4062. [CrossRef]